AD NUMBER

ADB018921

LIMITATION CHANGES

TO:

Approved for public release; distribution is unlimited.

FROM:

Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; FEB 1977. Other requests shall be referred to U.S. Army Missile Research and Development Command, Redstone Arsenal, AL 35809.

AUTHORITY

DRDMI ltr, 4 Sep 1980

TECHNICAL REPORT TD-77-1

VOLUME I – MIRADCOM/AFATL HYBRID SIMULATION FACILITY COMPATIBILITY STUDY

Aeroballistics Directorate
Technology Laboratory

8 February 1977

Sponsored by

Air Force Armament Laboratory
Armament Development and Test Center
Eglin Air Force Base, Florida

AFATL-TR-76-153

US Army Missile Research and Development Command
Redstone Arsenal, Alabama 35809

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: TD-77-1

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*: Volume I

MIRADCOM/AFATL HYBRID SIMULATION FACILITY COMPATIBILITY STUDY, Volume I.

5. TYPE OF REPORT & PERIOD COVERED: Technical Report

6. PERFORMING ORG. REPORT NUMBER: TD-77-1

7. AUTHOR(s): D. H./Dublin, K. L./Hall, W. M./Holmes

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Commander
US Army Missile Research and Development Command
Attn: DRDMI-TD
Redstone Arsenal, Alabama 35809

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 12 115p.

11. CONTROLLING OFFICE NAME AND ADDRESS
Commander
US Army Missile Research and Development Command
Attn: DRDMI-TI
Redstone Arsenal, Alabama 35809

12. REPORT DATE: 8 February 1977

13. NUMBER OF PAGES

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*: AFATL

15. SECURITY CLASS. *(of this report)*: Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

TR-76-153-Vol-1

16. DISTRIBUTION STATEMENT *(of this Report)*

Distribution limited to US Government agencies only; test and evaluation; February 1977. Other requests for this document must be referred to DRDMI-TI.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

DRDMI-TD-'77-1-Vol-1

DDC RECEIVED JUN 15 1977 C

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Simulation
Infrared simulation system
Electrooptical simulation system
Radio frequency simulation system
Missile simulation
Simulation language
Physical simulation
Missile systems

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report presents the results of a study conducted by the Advanced Simulation Center for the Air Force Armament Laboratory, Eglin Air Force Base, Florida. The requirement for this study was generated by the Air Force Armament Laboratory and had as its purpose an evaluation of methods and techniques whereby a high level of compatibility could be achieved between the two agencies in the development, implementation, and operation of hardware-in-the-loop

ABSTRACT (Continued)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

410 126

Block 20 (Concluded)

simulations. The study results were also to include a discussion of the costs associated with use of the resources of the Advanced Simulation Center by the Air Force Armament Laboratory.

8 February 1977

TECHNICAL REPORT TD-77-1

VOLUME I
MIRADCOM/AFATL HYBRID SIMULATION FACILITY COMPATIBILITY STUDY

D. H. Dublin, K. L. Hall, and W. M. Holmes

Sponsored by

Air Force Armament Laboratory
Armament Development and Test Center
Eglin Air Force Base, Florida

AFATL-TR-76-153

Aeroballistics Directorate
Technology Laboratory
US Army Missile Research and Development Command
Redstone Arsenal, Alabama 35809

FOREWORD

The study documented herein was requested and funded by the Air Force Armament Laboratory, Armament Development and Test Center, System Analysis and Simulation Branch, Eglin Air Force Base, Florida under MIPR 7621-76-90056.

CONTENTS

I. INTRODUCTION

A. Purpose

This report presents the results of a study conducted by the Advanced Simulation Center (ASC) for the Air Force Armament Laboratory (AFATL), Eglin Air Force Base, Florida. The requirement for this study was generated by AFATL and had as its purpose an evaluation of methods and techniques whereby a high level of compatibility could be achieved between the two agencies in the development, implementation, and operation of hardware-in-the-loop (HWIL) simulations. The study results were also to include a discussion of the costs associated with use of the resources of the ASC by AFATL.

B. Background

The US Army ASC consists of a centrally located, modern, hybrid computer complex surrounded peripherally by three environmental physical effects simulators. The infrared simulation system (IRSS), electrooptical simulation system (EOSS), and radio frequency simulation system (RFSS) are capable of spectral bandwidths and physical motions required for the evaluation of a wide variety of guidance systems and components. Under central computer control, the physical effects simulators (open and closed loop) provide real-time simulation capability, thus permitting precise and repeatable measurements of guidance system performance characteristics in nondestructive tests.

Realism is enhanced by controlled introduction of environmental factors such as meteorological phenomena, simulation of intentional electronic interference, and duplication of aerodynamic forces and counterforces. The three physical effects simulators operate in the microwave, infrared, visible, and ultraviolet regions of the electromagnetic spectrum. The ASC accommodates active, passive, and semiactive homing guidance systems, command-to-a-line-of-sight (CLOS), and command-to-a-point-in-space (CPIS) guidance systems in single or multiple modes (Figure 1). A detailed description of the ASC and its simulation capability are presented in Volume II.

Flexible structuring of the simulation elements provides single and multimode system simulations as well as subsystem, element, and basic hardware and man-in-the-loop evaluations. This feature, combined with "stand alone" computer simulation capability, provides simulation support from concept formulation to production and deployment. In addition to accelerating the research, development, test, and engineering phase of a program, ASC increases the level of confidence that a successful system will result and reduces program costs through optimization of time, minimization of flight testing, and early identification of potential false starts.

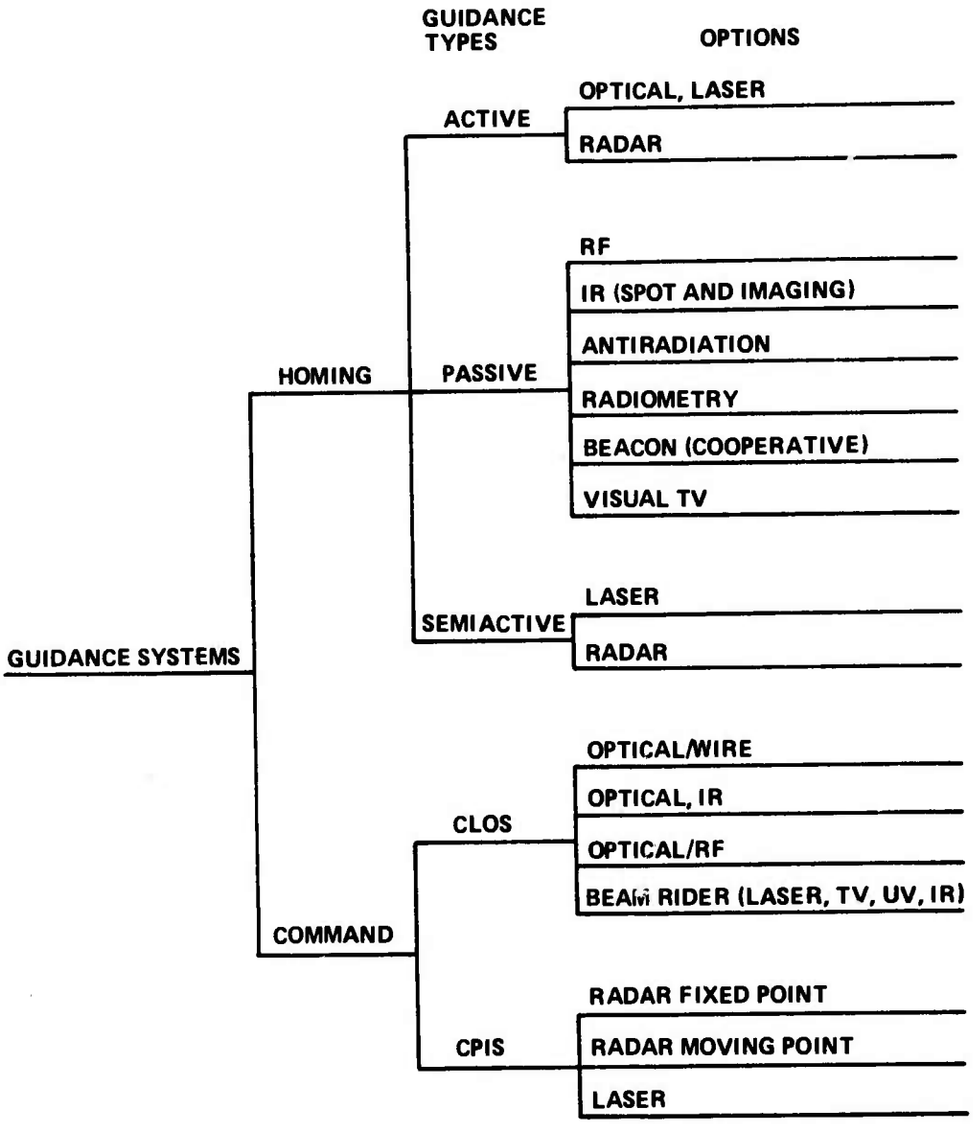


Figure 1. Simulation options available.

The AFATL has developed, independently, facilities for real time, hybrid computer, and HWIL simulations of guided weapon flight performance. This facility, although less sophisticated than ASC, has played an important role in Air Force development programs and will continue to do so as the AFATL development mission expands. The facility consists of a dual hybrid computer system to which is interfaced a five-axis motion simulator, an actuator test stand, and a target simulator. The target simulator rides the outer two axes, while the guidance system rides the inner three axes of the motion simulator. Utilization of the facility is heavy being used primarily in the area of detailed performance analysis as an integral element of development programs. In contrast to ASC, which primarily provides simulation services, the AFATL facility and staff provide simulation and analysis services with analysis being the primary function. In this regard, it has been instrumental in effecting early identification of potential problem areas and has contributed significantly to an increased success probability of test flights with commensurate cost savings.

As the development workload increases, it has beome evident to AFATL personnel that the facility must be upgraded and expanded to maintain effective and timely support to Air Force development programs. However, technological and economic uncertainties suggest that expansion to a capability similar to that of ASC should be a long term project. Furthermore, because much of the simulation work needed for weapon system analysis does not require the use of facilities like those of ASC, a satisfactory midterm solution is to use the ASC when needed and continue the use of in-house capabilities for the remainder of the work. Such a mode of operation is fully commensurate with the objectives of ASC because it was developed for use by all DoD agencies, and use by these agencies is encouraged.

For such a mode of operation to be realized in a cost-effective manner, it is necessary that the maximum possible level of compatibility exist between simulation development and implementation procedures at AFATL and ASC to ensure that potential cost savings are not diluted by the need to redevelop and implement simulations at ASC which are already operating at AFATL. It is to this objective that this report is directed.

C. Compatibility Issues

The basic objective when considering the question of compatibility between ASC and AFATL is to determine an approach whereby the cost and time to implement an AFATL simulation at ASC is minimized. ASC has established a well-defined procedure for simulation development and implementation which minimizes the "manual" effort involved. These procedures, although still evolving to some degree, have been shown to increase efficiency and decrease implementation time and cost and, at the same time, improve the quality of the finished product. For these reasons, it is unlikely that the established procedure will be changed

until a better procedure is developed. The procedure used by AFATL is much less formalized and is not readily characterized by a sequence of discrete, well-defined steps. Appendix A contains a general description of computers, hardware and software in use at ASC and AFATL. Appendix B presents an overview of the simulation development cycle presently being used at ASC.

The highest achievable level of compatibility between the two computer facilities would exist if digital computer programs and analog patch boards used at one site could be used at the other site without the slightest modification. This level is rarely, if ever, realized. In the present situation it can never exist as long as the computing equipment is so markedly different. Therefore, the attempt herein is to find an intermediate level which is practical and desirable.

In analyzing the compatibility problem, it is helpful to recognize that there are several "levels" of simulation which may be of interest. As described in Appendix B, ASC is normally concerned with development of a HWIL simulation but, in the process and integral to its normal procedure, an all-digital and a hybrid simulation are developed. These "lower level" simulation programs are free-standing simulations which are entities unto themselves and may be, for some projects, the end result. This has been the case for several projects which the ASC has undertaken. For the purposes of this study, it was presumed that the main interest of AFATL in achieving compatibility is to reduce cost and time for simulations which are beyond their capability, either because the simulation requires a capability which AFATL does not possess, or because the simulation requires computing capacity in excess of that available. Therefore, consideration has been restricted to transferring to ASC those portions of a HWIL simulation not concerned with control and operation of the simulation cells (this is a unique ASC function). No consideration is given to transfer of a digital or hybrid simulation (unless it is too large for the AFATL computer) because AFATL has the capability to do those without ASC assistance.

A customer desiring to use the ASC facilities can minimize cost and implementation time by accomplishing (at his own facility) some of the effort normally performed by ASC to develop and implement his simulation. A block diagram of the simulation process as followed at ASC is shown in Figure 2. Also shown on the diagram are seven customer entry (CE) points which represent those points where a customer may enter the process. At each point, it is assumed that all blocks prior to that point have been completed to the same level that they would have been at ASC. For example, if a customer enters at CE-1, all the effort would be performed within ASC. If a customer entered at CE-4, he would provide the following:

1) A statement of objectives.

2) An overall design of the simulation desired.

3) A complete math model of the problem.

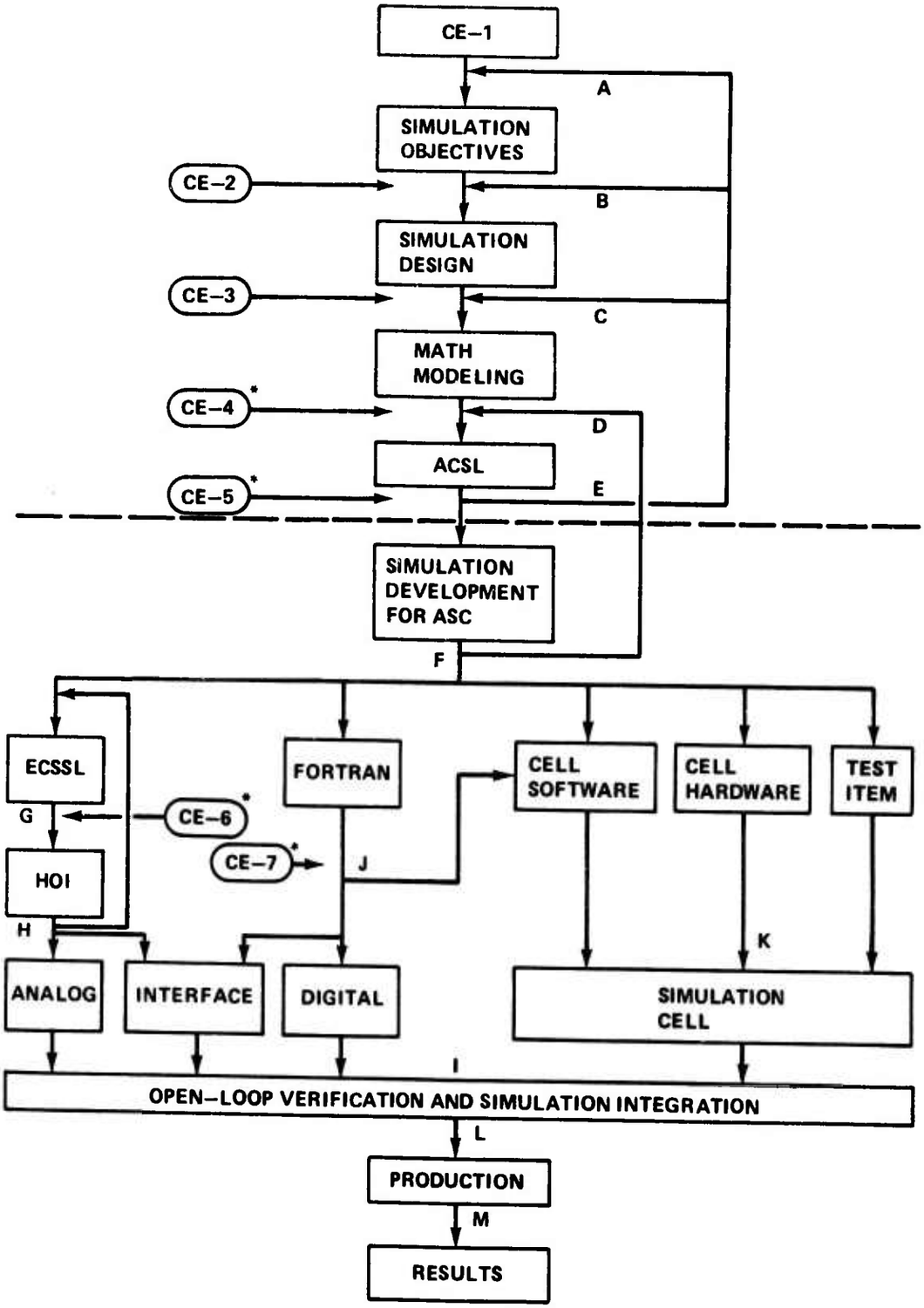


Figure 2. ASC simulation process.

An entry at CE-6 would require all of the preceding in addition to the following:

1) A verified digital simulation developed through advanced continuous simulation language (ACSL) or FORTRAN.

2) An error-free Extended Continuous System Simulation Language (ECSSL) compiler for Hybrid Computation run with resultant ECSSL output.

ASC requires strict adherence to documentation requirements at each stage of development. In the past, many problems have developed because of poor documentation.

Overall compatibility with ASC operating procedures will be determined largely by the degree to which customers choose to adopt similar procedures in their own facility and to use the hardware and software which produces outputs in the proper format to operate on equipment at both facilities. Section II of this report will present a discussion of the hardware and software changes which can be implemented at AFATL to allow entry at various points into the ASC process. Section III includes specific recommendations to AFATL on those changes which offer the most economic route to a large measure of compatibility.

II. HARDWARE AND SOFTWARE MODIFICATIONS AT AFATL

This section presents a discussion of several specific compatibility questions posed by AFATL in the statement of work. Specifically, the areas to be considered are as follows:

1) Feasibility and cost of a direct 6600-hybrid link at AFATL.

2) Dedication of present MICOM PACER to AFATL jobs or purchase of an additional PACER.

3) Use of ECSSL at AFATL.

4) Use of ACSL at AFATL.

In addition to these technical areas, which are major modifications or additions to the AFATL system, other less costly approaches for achieving a limited capability are discussed.

A. CDC 6600/Hybrid Computer Direct Link

This section presents a discussion of the "feasibility and cost of establishing a direct link between the Armament Development and Test Center (ADTC) CDC 6600 and the AFATL hybrid computers in order to achieve a hybrid computer complex at AFATL having a digital processing capability equivalent to the one at the US Army Missile Research and Development Command (MIRADCOM). The ensuing analysis

and discussion is predicated upon the presumption that the desired interface is one similar to or the same as the MIRADCOM Ports for Direct Digital/Analog Input/Output (PDDAIO) system. This system was developed to provide analog, discrete, and digital data transfer between the MIRADCOM CDC 6600 and the hybrid computers and dedicated minicomputers in the ASC simulation cells. It is concluded, however, that direct digital data transfer is probably not required because the physical simulation equipment is analog driven.

The technical problems associated with the design of a workable hardware/software system to interface the CDC 6600 to an analog/digital complex to operate in a time-critical, real-time hybrid mode have been fairly well resolved in the MIRADCOM PDDAIO systems. As with any new equipment design, some minor problems remain and are being solved as they arise. However, in applying this technology to the AFATL installation, additional problems of a technical nature arise due mainly to the configuration and location of the AFATL equipment. In particular, the following potential problems have been recognized:

1) Long distance separating the CDC 660C and hybrid complex.

2) Interfacing the 680/681 analogs to the CDC 6600.

3) Inability of the digital display system (DDS) terminal to operate over distances greater than 200 ft.

The greatest risk is associated with the first problem and is manifest in the effect that long distance and, hence, transmission delay time, has on the accuracy, rate, and volume of transmitted data. This transmission delay is caused by the physical reality that electrical signals propagate through any medium at a finite velocity (in wire cable) of approximately 0.5 ft/nsec which results in a delay of approximately 2 nsec/ft, one way. Interface hardware, by its very nature as a logic device, adds additional time delay into the overall transmission process. Many interface designs exist with many speeds and efficiencies, depending upon design features, technology, performance requirements, and physical constraints. In an effort to illustrate the effects of these transmission delays on interface system performance, a "typical" interface system is postulated. The performance range selected as typical is in the high average range to allow for possible improvement during the time required for procurement of the system. Its performance is characterized by a block transfer data rate of one data word every 800 nsec (assuming zero cable length). The analysis examines the effect of cable length, data block size, and frame time on performance of the system.

Essential to an understanding of this analysis is the concept of frame time[1]. The real-time operation with PDDAIO is an interrupt-driven

---

[1] The Society for Computer Simulation Definition of Terms for Analog and Hybrid Computers, 1 November 1975, SIMULATION, March 1976, pp. 80-86.

system wherein a cyclic interrupt is defined and generated either internally (software) or externally (hardware). On the occurrence of an interrupt, the CDC 6600 is called upon to perform an input/output process and a calculation sequence. The data input is used in the calculations and the calculation result is output. The time between interrupts is the frame time. This time is determined by the dynamics of the problem being run. Multiple interrupts may be scheduled in the system and are typically composed of high and low frequency subsystems. Typical frame times in the ASC ranges from 1 to 50 msec.

A complete description of the analysis program which was written is given in Appendix G. The analysis calculates the time to transmit a block of data and the percent of the frame time consumed as a function of cable length, data block size, and setup time. The total time required to transmit a block of data is given by:

$$TTXB = TTXD * NDWB + SUTM \quad ,$$

where

TTXB = total time to transmit block

TTXD = time to transmit one data word

NDWB = number of words in block

SUTM = channel setup time per frame.

The expanded version in this relationship is given in Appendix G along with the program output. The program was written in HYTRAN operations interpreter (HOI) and run on the PACER 100 in the ASC.

Using these relationships for two examples which approximate the upper and lower limits of performance achievable under the expected AFATL constraints (1000-ft cable length) the following results are obtained:

1) Example 1 -

   Frame time - 1 msec
   Data block length - 10 words
   Frame time consumed - 17%

2) Example 2 -

   Frame time - 50 msec
   Data block length - 100 words
   Frame time consumed - 3.5%

The primary effects on the overall system performance due to data transfer time are (1) time available for computation and (2) accuracy of transmitted data. These two effects are interrelated, i.e., high accuracy representation of wide bandwidth signals requires a large number of samples per cycle and, hence, short frame times. However, short frame times during which a sizeable number of words are transmitted means that not much computation time is left during the frame. The full analysis of this situation must be accomplished by the user who knows the exact job constraints which will exist.

Under these system guidelines, one can determine the maximum bandwidths of the simulation variables which can be transmitted between the facilities. If, for example, 0.01% accuracy is desired for a first-order system, then approximately 30 samples per cycle are required. The task is much more complex for higher order system simulation. A worst case example may require as many as 600 samples per cycle for a dynamic accuracy of 1% for an uncompensated simulation of a second-order system[2].

Thus, for the simple case of a first-order system simulation, an accuracy of 0.01% can be expected with 30 samples per cycle for a system of 30-Hz bandwidth using a 1-msec frame time or 1.7-Hz bandwidth for 20-msec frame time. But for a more practical situation, given 30 samples per cycle and a second-order system simulation, over 1% dynamic error can be expected for the 30-Hz bandwidth at 1-msec frame time and the 1.7-Hz bandwidth system at 20-msec frame time for an uncompensated simulation. Dynamic errors as great as 10% should be expected for these cases when only 10 samples per cycle are available. Individually compensated simulations can result in improvements on the order of 5 to 10 over the uncompensated cases.

The second potential problem is that of interfacing the 680/681 analog computers to the CDC 6600. This is not a technology problem because the same technology which has been applied to the 781 will be usable on the 680/681. However, an interface for this equipment combination has never been built. (Caution must be observed suggesting that, in MIRADCOM experience, the manufacturer of the analog equipment has had the most success in building an interface to his analog machine.)

The last problem is associated with the real time control terminal which operates under the DDS software. These terminals are cathode ray tube/keyboard terminals which were specially designed for use as remote terminals for control of real time jobs. The displayed video is generated at the CDC 6600 and transmitted over coaxial cable to

---

[2]Howe, R. M., Performance Analysis of Ultra High Speed Digital Function Generators in Hybrid Computation, 1976 Summer Computer Simulation Conference, Washington, D.C., 12-14 July 1976.

each terminal. It has recently been determined that the drive electronics for the video and the logic drive for the keyboard will not operate properly over cable lengths in excess of 200 ft. This presents no problem within ASC because maximum separation is only 150 ft. However, at AFATL the separation is much greater than 200 ft and, therefore, additional drive capability must be added. No insurmountable problems are foreseen because standard hardware can be used. However, it is pointed out here because the additional electronics represent added cost.

The cost associated with acquisition of a PDDAIO type real time system is high, and a realizable delivery schedule is easily perturbed. However, it is estimated that the system would cost approximately $2M and require about 30 months for delivery based on ASC experience with PDDAIO and making the following assumptions:

1) Already developed hardware and software designs will be used wherever possible.

2) Equipment is purchased through research and development channels rather than ADPE channels (hence avoiding the ADPE approval chain).

3) Task leader knows the technical approach and methods for expediting contractual actions.

Significant delays (up to 12 months) could result if any one of the preceding assumptions is violated. It must also be realized that the PDDAIO equipment is designed to operate only with a CDC 6600. Hence, if the digital computer at AFATL were to be changed during the acquisition of the real time hardware, the system must undergo almost total redesign, and the cost and schedule could double.

B. Dedication of MIRADCOM PACER to AFATL Jobs

This section presents a discussion of the practicality of dedicating a MIRADCOM PACER to AFATL jobs. The purpose of this approach is to minimize the software conversions required for the digital portion of a simulation. The use of a PACER operating with analog computers more nearly duplicates the equipment configuration at AFATL.

For a clear insight into the ramifications of this approach, one must differentiate between the types of jobs which may be desirable to transfer. For this purpose, two types are considered:

1) A hybrid simulation already configured to run on AFATL hardware which, for some reason, AFATL desires to run on ASC equipment.

2) A HWIL simulation which AFATL desires to implement on ASC equipment, the digital and analog portions of which AFATL will develop and transfer to ASC computers.

In the first case, very little difficulty is anticipated, provided that the digital portion of the simulation will fit within a single PACER. The analog portion will require recompilation and setup on the ASC analog equipment because the ASC analogs are different. However, as will be discussed in a later section, the compilation can be performed by AFATL through the use of the ASC version of the hybrid compiler. The ASC PACER has only 16 analog-to-digital converter (ADC) and 16 digital-to-analog converter (DAC) channels, and its interrupt structure and sense/control lines are different from a standard PACER system; therefore, the digital code would require some modification in order to run. Overall, the effort to convert is minimal.

For the latter case, however, the situation is not quite as straightforward and may not be economically or technically feasible.

Several factors hinder using an ASC PACER for Eglin simulations, some of which result from the basic design of the ASC hybrid/interface system. The ASC system is designed for the CDC 6600 to serve as the central control and digital processing device for an entire HWIL simulation. The high-speed data communication between the minicomputers in the cells and the simulation program in the other portions of the center can presently be best accomplished with the direct cell interface between the CDC 6600 and the minicomputers. In fact, the PACER is not presently interfaced to any of the cell minicomputers. A typical interface to either the IRSS or RFSS minicomputers would cost approximately $100 K. The cost would have to be borne by AFATL because ASC has no valid requirement for it.

The requirement for high data transfer rates emphasizes the problems associated with using the PACER in a HWIL simulation.

In the IRSS and the RFSS, high data transfer rates, short frame times (1 to 15 msec), and large numbers of data words (50 to 200 words) per frame are required to close the loop. In addition, in the RFSS most of the processing capability is digital, which increases the digital interface requirement. The RFSS and IRSS require transfer rates of approximately one word per 2 to 3 μsec. The RFSS datacraft equipment has a 24-bit word length and the PACER has a 16-bit word length. At the high transfer rates required, the PACER 100 direct memory access (DMA) channel would consume 50% to 100% of the PACER 100-cycle time during data transfer. The exact percentage will depend upon the complexity of the interface protocol.

If 50 words are transferred per 15-msec frame at 2 μsec per word, a small percentage (0.7%) of the total frame is consumed for data transfer. However, if 200 words are transferred per 1-msec frame at 3 μsec per word, 600 μsec of 1000 (60%) available microseconds are required for data transfer. The latter case is rather severe, but illustrates a worst case for the IRSS. A typical case would be 10 to 20% of the one 1-msec frame being used for data transfer in the IRSS and 5 to 10% in the RFSS.

If the digital portion of hybrid simulation to be implemented on the PACER has 5 to 20% frame time available after all other code is executed, this approach is technically feasible. However, a single PACER will not provide sufficient digital capability for most time critical applications in the ASC environment unless many simplifying assumptions are made and the entire program is written in assembly code.

Aside from these technical problems, an additional hindrance is the fact that the present PACER is used 1.5 to 2 shifts performing diagnostics, setup and checkout, documentation, scheduling, and costing. Any dedication of the PACER would curtail its use for these functions.

Based on these considerations, the only practical way in which an AFATL job could be performed using a dedicated PACER is through the purchase of an additional PACER with appropriate peripherals and interfaces. For a system comparable to the existing PACER system, integrated into ASC, the approximate cost is $330K. This amount is broken down as follows:

PACER 100 and standard input/output devices	$100K
High-speed printer/plotter and graphic terminal with hard copy and joy stick	60K
Hybrid linkage	110K
Interface to ASC trunking station and analogs	60K
Total	$330K

In addition, for HWIL operation, an interface to one or more cells at $100K each is required. The cost to convert the programs for a simulation can vary from approximately one man month ($5.3K) to about three man months (15.9K), depending upon the magnitude of the conversion required.

It is concluded on the basis of the preceding discussion, that from an economic and utility point of view, the dedicated use of a PACER to do HWIL simulation is not practical.

C. AFATL Use of ECSSL

For the last 2 years, ASC has used a hybrid compiler software package for automating the programming of analog computers. This package, APSE, was well developed and highly useful; however, it had problems primarily in the limitation of the types of analog computing elements which it could program. Furthermore, it only applied to an AD-4 computer. A new compiler, ECSSL, has recently been completed by EAI under contract to ASC. This new compiler incorporates enhanced operational features and will produce output applicable to the AD-4, EAI 781, and EAI 681 analog computers.

The ECSSL compiler is a language processor which translates scientific problem statements from a convenient mathematical form to an appropriate object language for hybrid computer solution. In addition to hybrid program generation, ECSSL can produce complete digital processor solutions of the problem. The same source program can be used to produce digital problem solutions, maximum and minimum values for scaling, the complete analog processor program, and a few high accuracy/resolution check solutions as the hybrid computer study proceeds. At a source language level, ECSSL interprets a superset of FORTRAN mathematical statements including facilities for specification of differential equations. Also, special data specification statements are included for initialization of differential variables and declaration of expected ranges of variables during problem solution.

Since the ECSSL compiler is written in standard FORTRAN IV, it operates on a variety of standard digital processing systems. Most processing is independent of the particular analog processor or hybrid computer on which the object program will run. A file of available parallel hybrid computing elements is automatically referenced for the generation of the object program. Therefore, the user can state his problem using the same source, independent of the particular analog processor upon which the job will run.

The object language of ECSSL is HYTRAN. This language is designed specifically for efficient automatic setup/checkout and executive control of parallel analog processors in Electronic Associates Inc. (EAI) and the AD/4 hybrid computing systems. The HYTRAN object program listing provides all necessary information to prepare the analog processor interconnection panels. Also, execution of the object program on-line in the hybrid system completely verifies the interconnections.

ECSSL also outputs data files for the generation of functions of up to three variables. Depending upon the physical devices available for function generation in the particular hybrid system, these files are used to automatically produce the necessary arbitrary functions.

Applications of ECSSL include physical system modeling or simulation, control system design, signal processing, training simulator programming, and general scientific/engineering analysis.

The installation of ECSSL on the CDC 6600 at Eglin is expected to present minimal problems because it is running on the CDC 6600 at MIRADCOM. Problems will most likely be restricted to training of personnel in the use of ECSSL and providing motivation to switch from old methods to the analog compiler. The latter has overshadowed all other difficulties at MIRADCOM, but the same difficulty has been experienced with other new products (assembler to FORTRAN, FORTRAN to simulation language, etc.). This motivation problem can best be solved by a positive policy requiring all simulations to use the compiler.

Actual cost of installation and initial training will be approximately $20K. This version would permit programming of AD4, 781, and 681 at Eglin, MIRADCOM, or other simulation laboratories. Future enhancements may vary in cost considerably, but after a new version of a hybrid compiler is developed, most costs are expected to remain in the $10K to $30K range for purchase of new versions. Development cost could run 10 to 100 times the installation cost. Other than cost, problems which can be expected in new versions will consist of normal debug for the very earliest issues and training on new features for optimum utilization of the new features. Minor modifications or updates, short of a new version, to the compiler can normally be made by local Eglin software personnel and should not cause significant cost or problem impact.

An analog program previously run completely through ECCSL and implemented in working condition on an Eglin 681 can be brought to a state of a complete static check on either the AD4 or 781 at ASC in approximately two man weeks. To convert a program running on the Eglin 681 but not compiled with ECSSL, to the AD4 or 781 to a state of complete static check in HYTRAN will require a minimum of ten man weeks.

D. AFATL Use of ACSL

The ACSL product package presently in use at ASC was originally purchased from its developer, Mitchell and Gauthier Associates, Concord, Massachusetts. Subsequently, this firm was engaged on a resident basis to provide support in installing the package, training ASC simulation engineers in its use, and developing enhancements to the package to make it better serve the needs of ASC. These enhancements concern, primarily, addition of multiple derivative sections and real time code as discussed in Appendix C.

AFATL is encouraged to contact Mitchell and Gauthier directly (AV 746-4141) to acquire information concerning procurement and installation of ACSL at AFATL. The package is now available through GSA contract GC-00C-00907.

III. CONCLUSIONS AND RECOMMENDATIONS

The analyses and discussions conducted during the preparation of the various sections of this report have enabled us to arrive at several relatively firm conclusions regarding the achievement of compatibility between ASC and AFATL in the development and implementation of "transportable" simulation programs. In addition, MIRADCOM is in a position to make some specific recommendations toward achieving this objective. These conclusions and recommendations are presented in the following paragraphs.

A. Conclusions

1) The potential exists for the achievement of a high degree of compatibility between ASC and AFATL.

2) Inasmuch as ASC has already achieved a stable operation using simulation development methods and software techniques, the most expedient approach to compatibility is for AFATL to adopt some or all of the techniques as part of their standard operating procedure.

3) A certain degree of compatibility in the area of analog programming already exists in that both facilities possess PACER systems and use HYTRAN software for setup and checkout of analog programs.

4) A direct link between the AFATL CDC 6600 and hybrid computers similar to the one at MIRADCOM is technically feasible. Such a link could provide high accuracy real time data transfer for combinations of frame time and block length as described in Section II. A. The cost of such a system would be on the order of $2M and require, as a minimum, 30 months for delivery.

5) The dedication of a MIRADCOM PACER, either at present or in the future, to AFATL jobs is not practical. For the limited capability realized in the ASC environment, purchase of a separate PACER for this application is not a cost-effective solution.

6) The adoption and use of the ACSL system would enhance simulation development at AFATL and aid in the achievement of compatibility. The most cost-effective means to transfer the digital portion of a simulation from AFATL to MIRADCOM is through the use of ACSL.

7) The adoption and use of the ECSSL compiler will further aid in the compatibility achievement. Furthermore, if AFATL implemented the Redstone version of HYTRAN and the core image generator (CIG) (Appendix A) on their PACER, complete software compatibility in the hybrid software area would be achieved.

8) The use of ACSL and ECSSL as a means of improving efficiency and quality in simulation developments is not an innovation of restricted applicability to large production-oriented facilities such as ASC; they are equally, if not more, useful in facilities such as AFATL, wherein system analysis is the primary objective because the time to realize an operating simulation can be shortened, and simulation development becomes a means to an end rather than an end in itself. By spending less time to develop simulations, an analyst is free to spend more time analyzing. As with any other "new way of doing things", the implementation of ACSL and/or ECSSL will require management support and emphasis.

B. Recommendations

To achieve compatibility between ASC and AFATL, the following recommendations are made.

1) AFATL procure and utilize ACSL and ECSSL as an integral part of the simulation development process.

2) AFATL obtain, from EAI or MIRADCOM, and install the Redstone version of HYTRAN and CIG.

3) AFATL adopt a standard procedure for modeling and simulation development. Such procedure should use available tools, such as ACSL, ECSSL, HYTRAN, or similar software packages to minimize the recurrent programming effort, improve the quality by allowing more thorough verification and validation of models, and provide accurate documentation.

4) AFATL personnel gain some experience in the use of ACSL, ECSSL, and the MIRADCOM system by bringing a well-defined simulation model of a system of AFATL interest to MIRADCOM and work through the programming of that model using ACSL and implementation on MIRADOOM equipment using ECSSL and HYTRAN. Arrangements to do this, if desired by AFATL, can be readily worked out.

5) AFATL give strong consideration to acquisition of a stand alone multivariate function generator (MVFG) to unload the PACER from this time-consuming process. Information on such MVFG can be obtained from MIRADCOM or EAI.

C. Suggestions

The material contained in this section is not provided in the form of specific recommendations, but rather as suggestions which will improve the operating efficiency at AFATL or further aid compatibility. It is suggested, therefore, that AFATL consider the following hardware and software acquisitions.

1) A Tektronix 4002A, or equivalent, graphics terminal as a peripheral on the PACER systems - Such a terminal, which has a refreshed scratch pad display, would allow the capability to use the edit capability of HYTRAN to change code on-line without having to reenter a complete line of code. Unfortunately, new 4002A terminals are no longer available, but one that is used might be located. As an alternate, a nongraphics refreshed-type terminal should be considered.

2) A remote dial-up terminal, consisting of a CRT terminal, line printer, and card reader to the MIRADCOM 6600 - Such a terminal could be procured for under $50K and, if matched by a similar terminal at MIRADCOM connected to the AFATL 6600, would permit generation of hybrid computer (HYTRAN), ACSL, and FORTRAN files at one installation from input at another. This mode of operation would accelerate the transfer of program information and eliminate extensive travel to transport programs.

3) A remote dial-up terminal to the MIRADCOM PACER for remote access to and manipulation of HYTRAN files - MIRADCOM has recently acquired such a remote terminal for its own PACER and could likewise access the AFATL PACER with the addition of a dial-up port. The acquisition cost of this terminal is under $15K, including the modem, and would require no additional software.

4) An analog diagrammer software package which is being developed at MIRADCOM - This software uses the HYTRAN file as an input and creates a multipage printed diagram of the analog program. The program available in the Fall of 1977 does not have complete interconnection capability, but requires the programmer to draw interconnects manually. The program, however, prints the identification of components to be interconnected. The hardware requirements to use this program are two removable platter disc systems and an electrostatic printer-plotter.

Appendix A. DESCRIPTION OF COMPUTER FACILITIES

1. ASC Hybrid Computer Complex

Figure A-1 depicts the ASC hybrid computer complex. This complex consists of four major systems:

a) CDC 6600 digital computer.

b) Ports for direct digital/analog input/output (PDDAIO).

c) Advanced simulation facility interconnect and setup subsystem (ASFISS).

d) Analog computers.

This system was designed under the philosophy that hybrid operations would be performed using the CDC 6600 as the digital element communicating through PDDAIO and ASFISS to the analog computers and simulation cells.

PDDAIO (shown in the upper left corner of Figure A-1) provides for real-time communication between the CDC 6600 and other elements of the center. Basically, it consists of 64 multiplying digital-to-analog converter (MDAC) and 64 ADC channels, a direct cell channel for each simulation cell, analog computer controllers, interrupt control and synchronization hardware, and the necessary interfaces to the CDC 6600 for real-time job control and data conversion. PDDAIO communicates to the analogs through the trunking station for data transfer and through the analog controllers and configuration switch for mode control. It communicates to the simulation cells through the direct cell interfaces providing direct digital data transfer. A set of digital display terminals is used for real time job control.

ASFISS is designed to provide a central tie point for all elements of the center. The hardware for the ASFISS system includes the following major subsystems as shown in Figure A-1:

a) ASFISS trunking system.

b) ASFISS controller and peripherals.

c) Multivariate function generation (MVFG).

d) Hybrid control hardware.

The trunking system allows interconnection of the elements of the center through a set of 2304 analog trunks and 960 discrete trunks (expandable to 4096 and 2048 lines, respectively). The trunks are patched via interconnecting cables which patch trunk lines in groups of 16. Each cable connector contains a 22 pin block for carrying the

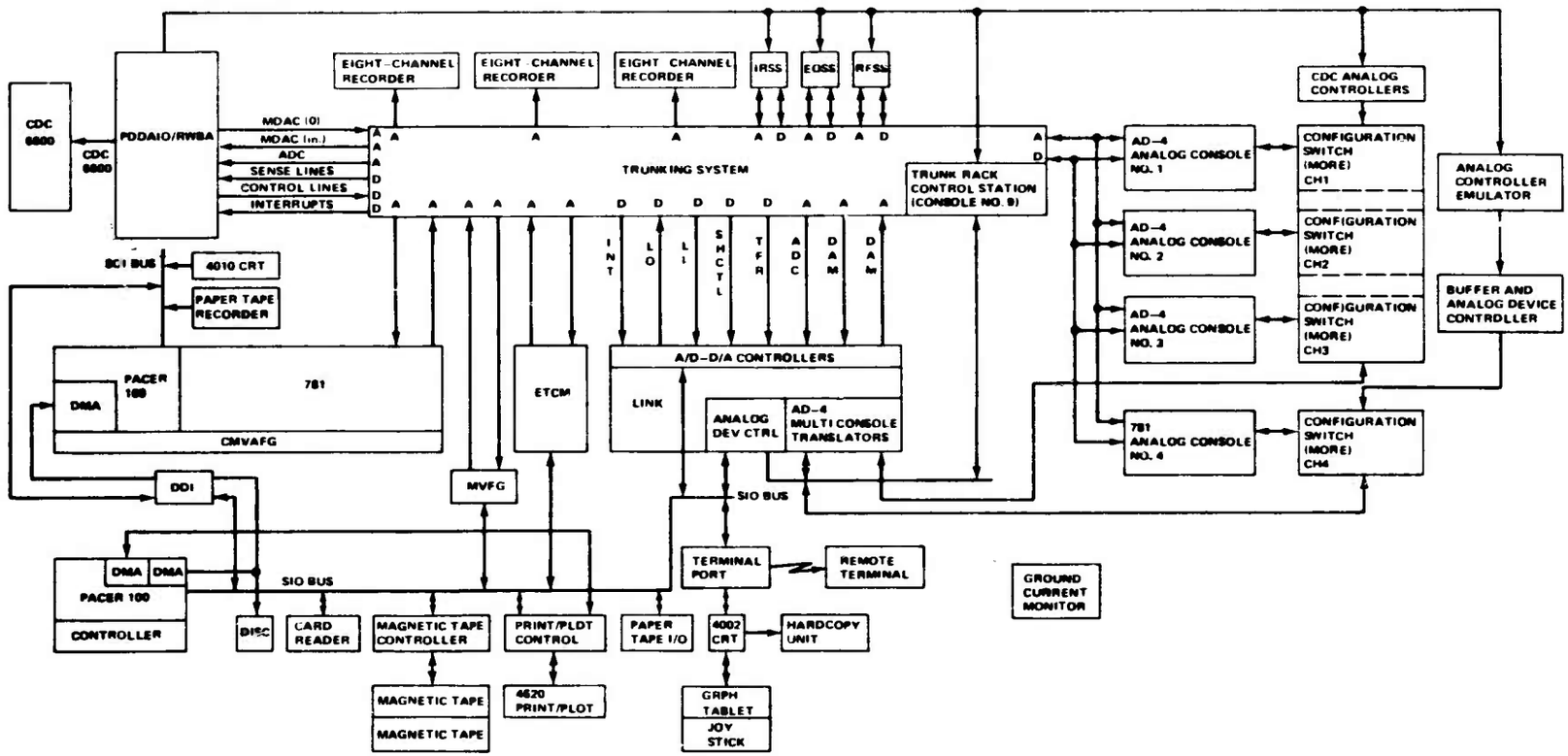


Figure A-1. ASFISS system block diagram.

connector code. Analog buffer modules are provided for buffering signals into and out of the trunking station. A control station provides a capability to address all trunk lines for patching status and value readout and linkage to the ASFISS controller (PACER 100) or to PDDAIO for trunk status and value readout.

The ASFISS controller is a PACER 100, 16 bit, 32K core minicomputer with peripherals as shown in Figure A-1. The system is a disc-oriented system using removable cartridge disc for primary system bulk storage. The magnetic tape units are seven-track devices for compatibility with CDC tape systems. Primary control of the system is through the 4002 graphic display terminal. The controller also includes a direct digital interface (DDI) to a second PACER 100 associated with the central multivariate arbitrary function generator (CMVAFG) for communication of function data which has been processed by the controller PACER. The electronic trunk control module (ETCM) is an experimental 128X64 analog switching matrix for switching trunks under control of the controller PACER.

Centralized arbitrary MVFG for the center is provided by two subsystems. One subsystem is the CMVAFG, which consists of a 781 analog computer, a digital coefficient attenuator (DCA) expansion rack, and a PACER 100. This system is configured to be loaded with function data from the controller PACER and this operates as a central resource during a simulation. The analog generates the output function with break-point configuration controlled by the associated PACER. Analog input/output is via trunk lines through the trunking station. The only peripherals provided on the PACER are those required to set up the function generator. The CMVAFG can generate up to 15 functions of one variable, 10 functions of two variables, or 5 functions of three variables. The second subsystem is the MVFG, a second generation version of the CMVAFG, which is connected to the Controller PACER input/output bus for setup and function data loading. Analog input/output is via the trunking station. This function generator provides approximately 2 to 4 (depending upon the application) times as much generation capability as the CMVAFG.

The hybrid control hardware consists of the various analog controllers, a set of 16 ADC/MDAC channels, and the configuration (MORE) switches. The configuration switches are multiple OR switches controlled from either the controller PACER or PDDAIO (determined by the setting of a manual switch), which switches the analog computers between the CDC 6600 analog controllers and the ASFISS analog devices controller. This allows either the CDC 6600 or the controller PACER to have mode and setup control over the analog computers. The analog controllers provide hybrid interfaces between the digital computers and the analogs for mode and setup control.

The analog computer pool presently consists of three AD/4 and one EAI 781, ±100-V analog computers. Future expansion will add two EAI 781 machines. The analog and logic trunks are connected to the trunking station.

The PACER 100 (designated as the ASFISS controller) is provided to the system primarily for rapid and effective preparation, setup, verification, and documentation of hybrid simulation models and was not intended to perform the digital function in a hybrid simulation; this function is reserved for the CDC 6600. This function is accomplished using a set of specially tailored software including enhanced standard EAI software and specially designed software. At the heart of the ASFISS PACER software is HYTRAN. In addition to its role as a supervisory routine for the ASFISS software system, HYTRAN provides the system operator with a means of setting up, editing, verifying, and operating the hybrid simulation. The capability of HYTRAN has been greatly enhanced for the ASFISS application. First, HYTRAN programs need not be entirely core resident; they may be swapped in and out of a disk file thus allowing almost unlimited program size. Also, the interface between HYTRAN and FORTRAN programs has been made more convenient through improvements in argument transfer and data storage techniques. Further, a separate HYTRAN core image is no longer necessary for each group of FORTRAN routines. "LOAD" and "SWAP" routines have been added to allow run time program loading or HYTRAN/program swapping. Several input/output interface modifications have been implemented to allow HYTRAN to more effectively use the disk, seven-track magnetic tape, and the 4002A Tektronix display. Also, HYTRAN has been given the capability of accessing the 16 ASFISS logic input/output lines and the trunking station logic and voltage values, and verifying the interconnection status of the trunking station connectors. Finally, a new directive has been added (WLDEX) which allows HYTRAN to load a core image file over itself and execute the loaded program. HYTRAN uses this directive to control the loading and execution of the function generation processors.

The function generation processors provide the ASFISS user with the means for creating, editing, storing, and processing multivariable functions. As mentioned previously, the function generation processor is executed by means of the WLDEX directive in HYTRAN. HYTRAN may be reexecuted by an HOI directive in the function generation processor. Functions may be created by the user and stored on disk to be run at a later time on the CMVAFG or the MVFG. The function descriptions may be defined at the keyboard or they may be entered via the card reader from seven-track magnetic tape (generated by ECSSL or from cartridge disk files). Once the user has defined the functions desired, the functions can be assembled and automatically generated in the function generation hardware.

The standard overlay CIG has been enhanced to provide a powerful tool for producing system overlay modules and also a convenient means for the system user to produce and communicate with external program overlays to be used in conjunction with HYTRAN programs. The enhancements of the standard CIG are in the following three areas:

(a) An overlay capability

(b) The ability to output memory map files to the disk

(c) The ability to produce a sorted memory map.

Two other software packages developed for ASFISS are the DDI software and special diagnostic programs. DDI is required to allow the control PACER to communicate with the CMVAFG PACER to execute function generation programs in the CMVAFG machine.

The special diagnostics include diagnostic routines for the trunking station, the CMVAFG, the MVFG, the ETCM, the AD/4 analog and 781/AD/4 translator, and the printer/plotter to facilitate maintenance and support of this special purpose hardware.

2. AFATL Hybrid Computer Complex

A block diagram of the AFATL hybrid computer configuration is shown in Figure A-2. This complex consists basically of dual PACER 600 hybrid computers with the analog consoles intertrunked to each other and to the simulation equipment. The EAI 693 equipment is a hybrid linkage between digital and analog computers providing for analog mode control and data interchange via ADC and DAC channels. Each PACER system represents an autonomous hybrid computer. Two analogs can be controlled directly from each PACER 100 with two additional analogs slaveable via analog and logic trunks. One PACER 100 has (through the DDI) an EAI 640 digital processor attached to provide auxiliary computing capacity. Each PACER is associated with a set of assorted peripherals. In the AFATL system, the PACER 100 digital machines are used as the digital elements of hybrid simulations to perform the functions of analog control, digital computation, data transfer control, function generation, etc.

The software system which supports the AFATL complex is the disk operating system (DOS), a disk resident executive software system. The DOS provides a number of application-oriented programs including an assembler, a FORTRAN compiler, and the HOI. The HYTRAN processor provides for hybrid simulation setup and checkout. It is used mainly by AFATL for performing status checks of analog computer programs. Because of the fairly limited core storage capacity, the digital portions of hybrid simulations are generally written in assembly language to minimize memory requirements and maximize execution speed. No high-level simulation languages are employed.

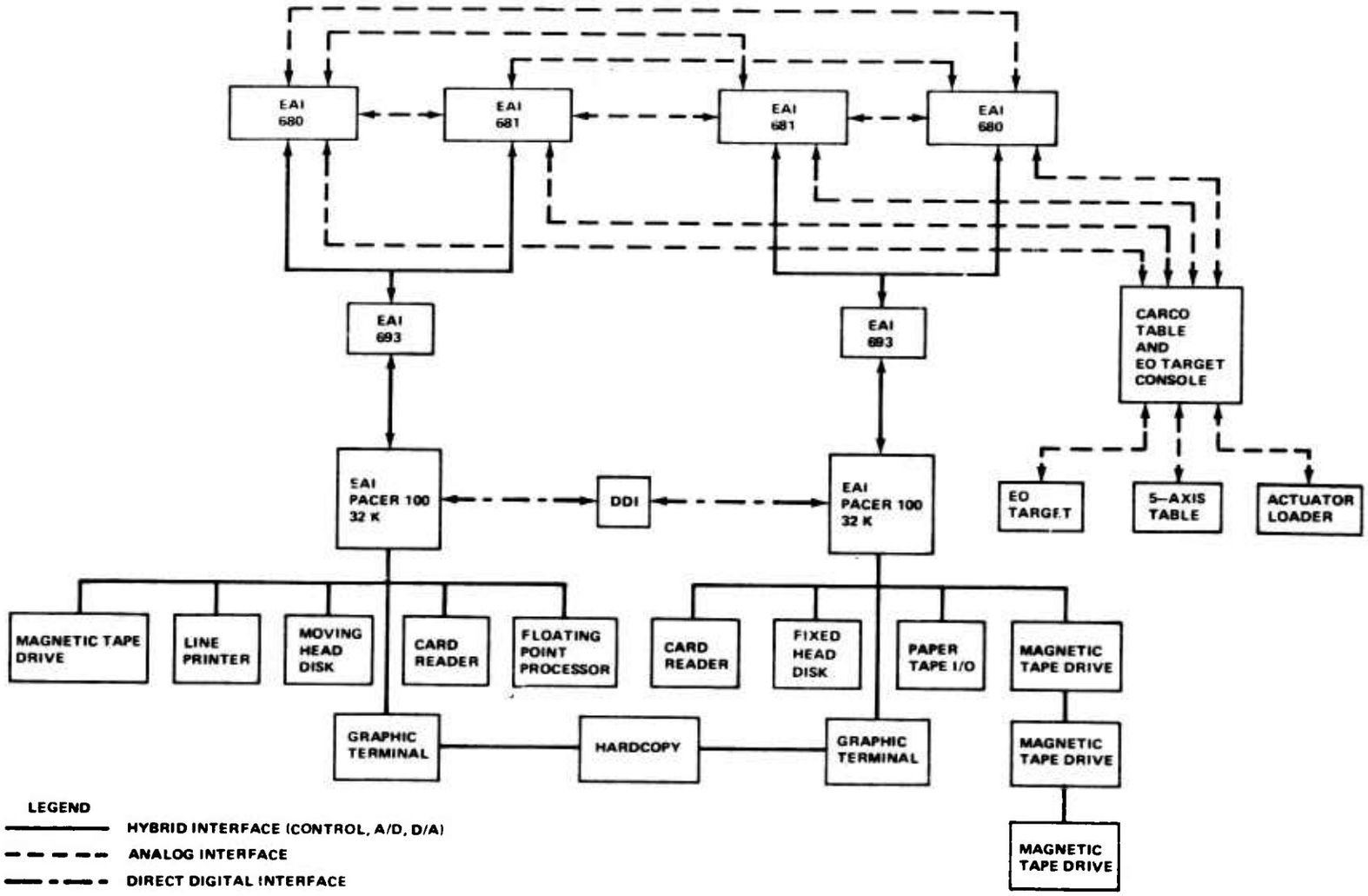


Figure A-2. AFATL hybrid computer complex.

Appendix B. CLOSED-LOOP SIMULATION DEVELOPMENT IN THE ADVANCED SIMULATION CENTER

1. Overall Development Cycle

Closed-loop simulation is defined here to include any simulation effort that uses combinations of the environmental effect simulation cells (IRSS, EOSS, RFSS), analog computers, or the CDC 6600 digital computer. This also includes closed-loop guidance in the normal guidance and control concept. The guidance loop may be closed analytically in an all-digital simulation or by the inclusion of specific hardware subsystems in a time critical simulation, i.e., autopilot, guidance computer, sensor, etc. Open-loop testing refers to the testing and data gathering to characterize specific hardware subsystems. This can be accomplished in the individual ASC cell as required or performed at other facilities. While the major portion of any required data base may be generated at other locations, a minimum of hardware characterization is typically required in the particular cell to familiarize personnel and to verify operation when equipment is received in the cell.

Experience in processing simulation tasks in the ASC has established a number of essential steps necessary to obtain desired results, i.e., achieve the objectives of the simulation task within established milestones at minimum cost. The major operational areas identified in the ASC closed-loop simulation development includes:

a) Establishing simulation objectives.

b) Performing simulation design and analysis.

c) Developing system and related subsystem mathematical models.

d) Developing a digital computer program using the math models.

e) Hybrid computer simulation development.

f) Developing HWIL operation.

While not all simulation will necessarily include all these areas, all simulation will progressively evolve through this sequence until the simulation objectives have been achieved. However, within each of the operational areas, a large number of other task-dependent operations are involved. Additional details will be discussed using the generalized event chart in Figure B-1.

2. Operational Areas

a. Simulation Objectives

The initial effort in this area is to identify specifically what will be the final use of the simulation and who will be using the simulation. Will the simulation be used to perform analytical,

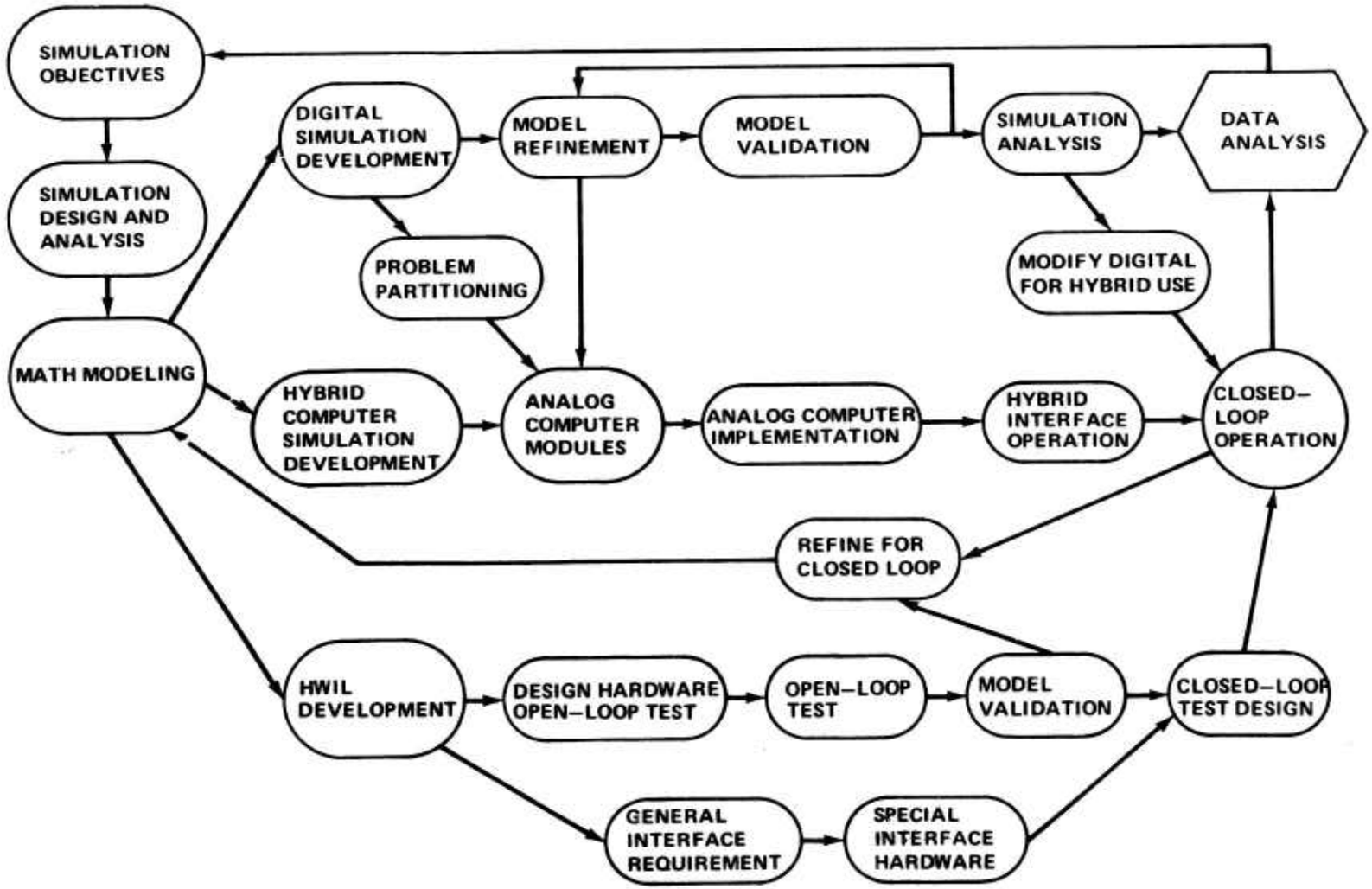


Figure B-1. Generalized event chart.

statistical, or concept studies, exploratory development with prototype hardware available either as a data base generator, or for HWIL operations or, will the simulation be used as support to a flight test program requiring HWIL operation with the maximum number of hardware subsystems?

The general level of model validation expected should be identified and consistent with the initial objective.

b. Simulation Design and Analysis

The initial effort during this area of activity is to establish, as a minimum, the nucleus of an ASC project team as depicted in Figure B-2. Team members include customer representatives, ASC members, and specialists as required. As a "minimum," two ASC members would be assigned on the team. A representative from the simulation modeling and analysis group and, depending on the nature of the problem, a representative from the hybrid implementation group or from the particular environmental effects cell involved. The team nucleus is expanded to meet the requirement and objectives of the simulation development.

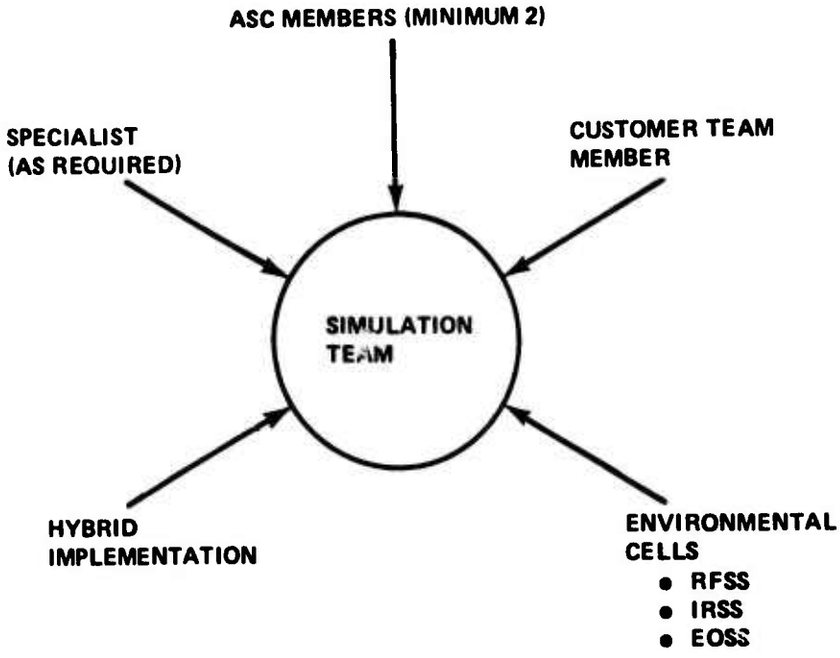


Figure B-2. ASC project team.

A second major activity in the simulation design and analysis operational area is to identify available data bases generated from previous simulation and analysis activity and related real world systems. Particular attention is given to open-loop test data on hardware subsystems, existing mathematical models, and simulations with the related set of assumption and hypothesis associated with the models. In particular, what is the quality of existing models as related to the present simulation objectives and the real world system? The logical structure of the simulation is established during this area of activity. Special hardware requirements are identified, i.e., interfaces, system hardware, special purpose equipment, etc. General software requirements are outlines, e.g., statistical data packages required for large scale data analysis, simulation language and related programming operation, applicability of modular missile simulation programs, and transportability of existing simulation programs from the customer to ASC facilities. In addition, general guidelines are established for open-loop and closed-loop test requirements on subsystem hardware to be included for HWIL operation.

Specific milestones are established for achieving the deliverable end items of the simulation task. Coordinated with the accomplishment plan is the scheduling of ASC resources, i.e., cell, analog computers, and hybrid interface utilization as available in conjunction with other on-going projects.

c. Mathematical Modeling

The mathematical modeling activity is directed toward describing the dynamics and related operation of the real world system. This typically progresses on a subsystem-by-subsystem basis. A typical missile system may be viewed as a modular structure as shown in Figure B-3. Available models are examined for uniformity in fidelity consistent with the objectives of the simulation, i.e., analytical, HWIL, etc. Available real world systems data are examined for model comparisons and model development.

Analysis is performed on existing models to gain insight into the dynamics of the particular subsystem. Particular information is required on frequency response, nonlinear operation, figure-of-merit for comparison of data bases between real world systems, and math models. A number of techniques are used for "goodness-of-fit" for data base comparison for graphical comparison and data analyses. Typical techniques used for graphical or "approximate" comparisons are: number of turning points, timing of turning points, direction of turning points, amplitude of fluctuation for corresponding time segments, average amplitude over the whole series, exact matching of values of variables, probability distribution, and variation about means. A more detailed goodness-of-fit on the data is obtained from data analyses using such techniques as: analysis of variance, regression analysis, spectral analysis, chi-square test, nonparametric test, factor analysis,

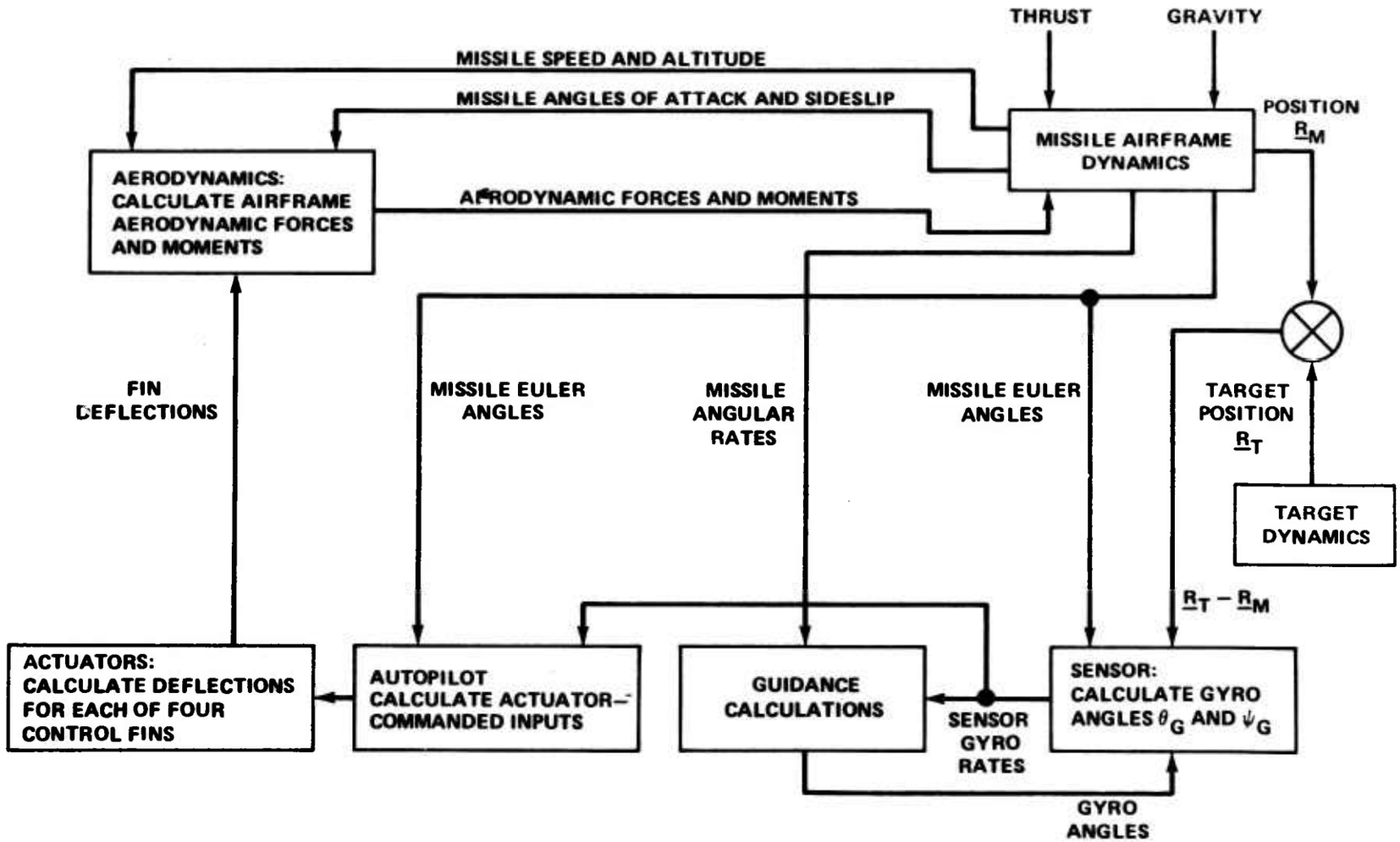


Figure B-3. Overall mathematical model (missile system).

Kolmogorov-Smirnov test, etc. In the absence of models or real world data, approximate models are established for each subsystem module. Probabilistic models require that means, variances, and distribution be identified which characterize the real world system.

The mathematical modeling and analysis typically involves the use of existing digital computer programs such as generic modular missile programs or specially developed programs for each subsystem using the ACSL program.

d. Digital Simulation Development

The development of a comprehensive digital computer program is considered essential for all simulations: pure analog, combined analog-digital hybrid, or hybrid with HWIL operation. The digital program may be initiated as a new program or, where possible, use generic simulation models and change the simulation to reflect the new models. A simulation language-based approach is used in the ASC for digital simulation development. This approach reduces cost and lead time and provides for easy development of generic simulation models with easy modification for quick-look and concept development. The digital program is used to check the mathematical model developed for the real world system.

The ACSL (pronounced AXLE) is used for digital simulation development in the ASC. The basic ACSL package has been expanded to include multiple derivatives sections and a hybrid real time operating capability. The ACSL single derivative option is used for the development of a standard digital simulation using a single step size and integration algorithm. A summary description of the basic ACSL system is presented by Holmes[3]; more detailed information is presented by Mitchell and Gauthier[4].

Verification of the digital computer implementation of the mathematical model is an essential step in simulation development. Verification, as used here, is concerned with the establishment of the correctness of a model. This includes a test of the correctness of computer coding used in the model, tests to determine the accuracy or correctness of the assumptions and hypotheses upon which the model is based, and testing the agreement of observed outputs and model prediction when the

---

[3]Holmes, Willard M., Time-Critical Simulation Development and Operation Using a Real Time Oriented CSSL, 1976 Summer Computer Simulation Conference, Washington, D.C., 12-14 July 1976.

[4]Mitchell, E. E. L. and Gauthier, J. S., "Advanced Continuous Simulation Language (ACSL)," Simulation, March 1976, pp. 72-78.

model is run, using as inputs the data employed in the construction of the model. The verification process is completed before the validation process can be effectively initiated. Validation, as used here, is an iterative process primarily concerned with determining the usefulness of a model as a representation of the physical system. For this purpose, validation data must not have been used in model development and the data must be of sufficient precision to make the test meaningful. The objectives of the modeling exercise must be kept in perspective since all data are not appropriate for use.

The verified digital programs and validated models are used as a reference in generating a data base for comparing results with programs involving multiple derivatives. This logically extends to providing a significant step in the verification process for analog and hybrid computer simulations.

e. Hybrid Computer Simulation Development

Given that a hybrid computer simulation has been established as part of the simulation objectives, the all digital program will be structured differently during the program development. Partitioning the mathematical model between the analog and digital computers is a major task in developing a hybrid computer simulation. A digital simulation of the hybrid computer operation is a cost effective way to study the effects of partitioning, frametime compensation, and a means of generating dynamic check data and scaling information for analog computer implementation. For a modularly structured simulation, each simulation module can be associated with a major subsystem in the real world system. Mathematical model fidelity as required for computer simulation and HWIL operation can be studied.

ACSL has been extended to include a multiple derivative capability to more readily simulate hybrid computer operation. This permits the models which are to be implemented on the analog computer to be represented in a single derivative section using an integration step size and an algorithm independent of other derivative sections. Other derivative sections can be used to represent the digital portion of the hybrid or subsystem operations to be replaced by the physical system hardware.

The baseline digital program is used as a reference to evaluate the effects of problem partitioning using the multiple derivatives. When satisfactory results are achieved, the mathematical models associated with the analog computers are prepared for hybrid compiler processing using the EAI-ECSSL compiler. The output of this compiler is a scaled analog computer diagram with assigned analog computer components and patching diagrams. Detailed procedures involved in using the hybrid compiler are presented in Appendix D.

An essential step in the verification of the analog computer implementation is a comparison of the response of implemented analog modules with digital results. This subsystem comparison approach is extended to include closed-loop system response for digital and hybrid operations. Typical comparison of digital and hybrid results are shown in Figure B-4.

f. HWIL Operation

If operation of the simulation with system hardware in the loop has been established as an objective, the cell involved in the simulation, i.e., IRSS, EOSS, or RFSS must perform the necessary effort to prepare the cell for the simulation. This effort encompasses a wide range of activities but can generally be categorized into three areas:

1) Environmental model and software development.

2) Test and interface hardware/software integration.

3) Open-loop test planning and implementation.

Once all the cell-related activities have been completed, the simulation team then becomes involved in the overall integration and verification phase of development.

The modeling and software development effort is concerned with generating target, background (clutter), and electronic countermeasure (ECM) models which will be used in the simulation, and requisite software programs to run on cell computer to drive the equipment to produce these models in the cell. In some cases, the software will be a part of the main simulation program. Effort is also required to verify that the desired environment is being generated within the cell. This usually involves a series of tests.

The integration of hardware and software is also a multifaceted effort. It includes such activities as:

1) Designing, fabricating, and testing mechanical and electronic interface hardware to interface the system hardware into the cell for operation, control, and monitoring.

2) Designing and implementing any special interface equipment to connect the cell hardware to other hardware elements in the center.

3) Ensuring that the cell software and computer operate/communicate properly with central computer facilities.

Open-loop hardware testing is necessary for several reasons. The cell operating personnel must be trained in the operation and control of the test hardware. Data from the open-loop tests will be used to verify proper operation of the hardware. More importantly, the

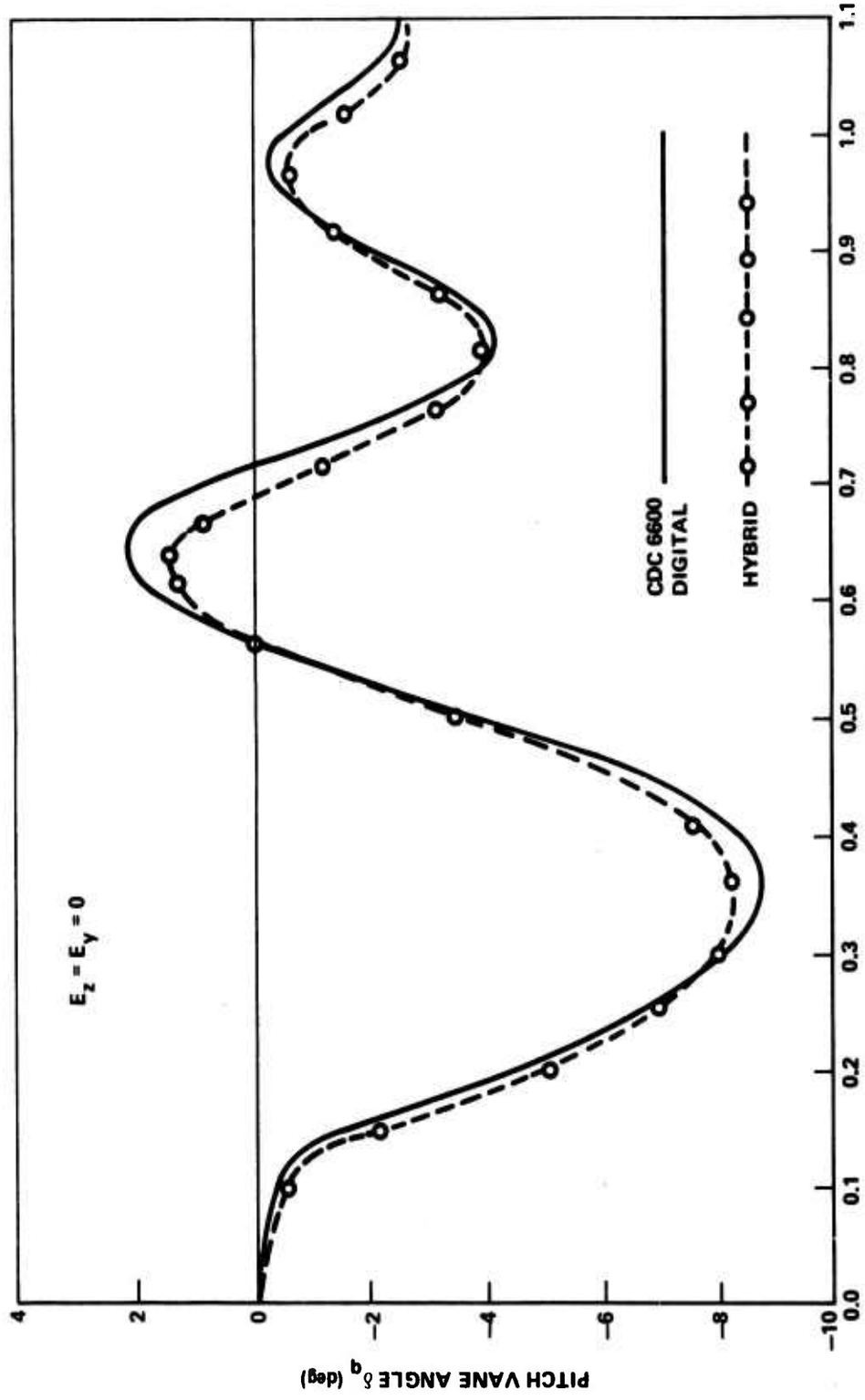


Figure B-4. Comparison of pitch vane angle δ_q from digital and hybrid programs.

open-loop test data will be used to verify and validate the correctness of the math models which have been developed for the hybrid simulation program.

The final step in the overall simulation development effort is the integration and checkout, via dynamic real time runs, of all elements involved in the simulation. Once an operational simulation is achieved, the validation process is initiated. This is accomplished in an iterative fashion wherein simulation-derived data are compared with test data from a comparable or the same scenario and any differences resolved by modifying and/or updating the models used in the simulation. This process is continued until the simulation reproduces test results to the degree of fidelity commensurate with the intended use of the simulation. Test data used in this process may originate from a number of sources such as flight data, wind tunnel tests, component testing, etc. Since the first step is a validation of the entire simulation, flight test data are the most valid data for comparison. However, if flight test data are not available, then other data must be used. In these instances, the various components of the simulation may be validated against data which apply to that component. The validation process, however, is incomplete until the complete simulation can be compared with complete system data.

Appendix C. ADVANCED CONTINUOUS SIMULATION LANGUAGE

1. Description of the Language

The ACSL system is designed for modeling the behavior of continuous systems described by time-dependent nonlinear differential equations and transfer functions. Typical areas of application are control-system design, electrical circuit analysis, missile and aircraft simulation, and fluid-flow and heat-transfer analyses. Program preparation can be from block-diagram interconnections or conventional FORTRAN statements or a mixture of both.

Highlights of the language are its macro capability, independent error control on each integrator, free-form input, and generation of functions of up to three variables. Many simulation-oriented operators such as variable time-delay, dead zone, backlash, and quantization are included and made readily accessible.

Macro capability allows constructing representative blocks (e.g., a transistor, an actuator, or an element in a heat-transfer problem) for systems containing state variables. Access is possible to any FORTRAN subroutine either predefined in a library or included in the ACSL model definition.

The powerful array capability allows breakdown of partial differential equations into sets of ordinary differential equations. Macro operations can pick up array dimensions so that size changes can be kept in one area in the simulation program. An array integration operator complements this facility so that vector and matrix integrations can be set up with a single statement.

Independent error control on each integrator allows faster execution by relaxing the accuracy specification on high-frequency variables. At the end of each simulation run, the variable-step integration routine reports the effect each state variable had on controlling the integration step size so as to allow appropriate action to be taken. The relative error and absolute error allowable can te specified individually for each state variable.

The basic structure of ACSL follows the specification established by the SCI Technical Committee on continuous system simulation language (CSSL).

The simulation program is defined in a model-definition section which consists of four parts:

a) INITIAL - Statements that are performed once. Typically, these statements lead to the calculation of initial conditions on the state variables; e.g., launch angles of a missile which depend, after a fairly lengthy calculation, on target position and velocity.

b) DYNAMIC - Statements that are performed at every communication (data output) interval and whenever the data recording operation takes place. A simple example would be the conversion of radians to degrees. Efficiency is improved if those calculations relevant to data recording operations only are collected into one block and performed together. Fixed time-step operations (as in sample-data systems or digital-computer control algorithms) can be modeled in this section.

c) DERIVATIVE - Describes the calculations that determine the derivatives of all state variables. This section allows the integration routine selected to advance the state of the system with respect to its independent variable, usually time. The statements in this section need not be ordered but will be automatically sorted into the correct sequence so that intermediate values are calculated prior to their use.

d) TERMINAL - Statements that are performed once at the end of the simulation run; e.g., mean and standard deviation calculations for Monte Carlo sequences and the radial miss-distance of a missile from a target.

The four integration algorithms included in the system are as follows:

a) Runge-Kutta Fourth Order (RK-4) - A dependable fixed-step method that may not be the fastest but has almost no trouble with stability, provided the integration step size is chosen sufficiently small.

b) Runge-Kutta Second Order (RK-2). Similar to the Fourth Order but not as accurate. For some systems with small controlling time constants, this can be almost twice as fast as RK-4.

c) Adams-Moulton Variable Order, Variable Step - As implemented by Gear, this algorithm chooses the order (from one to six) and step size so as to take the largest step commensurate with satisfying the specified error criteria.

d) Gear's Stiff-Variable Order, Variable Step (Implicit) - This method, also implemented by Gear, is particularly suited to modeling systems that contain characteristic roots that differ by many decades. When the higher modes have stopped, the algorithm can take steps many times times larger than the shortest time constant, an operation not possible with any other method.

Unlike other CSSL languages, no limit exist for any of the internal tables because they can grow to fill the available core space. Most simulation languages establish arbitrary limits on such things as number of symbols, number of state variables, labels, etc. These table sizes are typically built into the translator at compile time and can only be changed by system programming personnel updating the source cards and recompiling. In ACSL, no such artificial limits exist. As

a corollary, small programs can be executed in a smaller field-length because the simulation program does not have to be configured to accommodate the largest conceivable program any user may submit.

ACSL has been extended to include a multiple derivative capability to more readily simulate hybrid computer operation. This permits the models to be implemented on the analog computer to be represented in a single derivative section using an integrating step size and algorithm independent of other derivative sections. Other derivative sections can be used to represent the digital portion of the hybrid or subsystem operations to be replaced by the physical system hardware.

The modified ACSL package places the multiple derivatives in the dynamic region. The specifications for multiple derivative operation are shown in Figure C-1. The actual order in which the derivative sections are executed can be based on a number of criteria. For the example shown here, the execution of the multiple derivative simulation is driven from elements in an event list based on which derivative section is furthest behind in time.

Consider an example where the problem has been partitioned for two derivative section operations. The D1 derivative section represents the digital portion of a hybrid computer simulation with a frame time of 10 msec and the D2 derivative section represents the analog portion of the hybrid computer simulation with a frame time of 2 msec. The sequence of execution events can be described using Figure C-2. The order of execution at T = 0 is as specified in the simulation model, i.e., D1 is executed before D2. D1 is advanced over the 10-msec step to E1 (Event 1). D1 information regarding time and step size is reentered in the event list. Data required for D2 are available at event time E1 (10 msec). D2 is advanced over a time step of 2 msec. D2 information regarding time and step size is reentered in the event list. D2 is behind in time, therefore D2 is advanced again to E3, reentered in the event list, compared, found to be behind, and is advanced to E4. This sequence of operation continues until E6. At this time, D1 and D2 have been advanced to approximately the same point in time, and data would be transferred to D1 for advancement to E7. However, the numerical accumulation of time and round-off error can contribute to conditions that result in other than an exact numerical comparison of time. Consider that the accumulated time associated with D2 and entered in the event list is slightly smaller than time associated with D1. D2 will advance to E8, time is entered into the event list, then D1 is the furthest behind and is advanced one frame time to E7. Data are transferred backward in time for D2 to advance to E9. This seemingly random order of execution takes place only at the first alignment of derivative section. The conditions that prevail at that point repeat for future points in time. These phenomena have an averaging effect and are similar to the calculated derivatives using a Runge-Kutta type algorithm. Without special considerations, the derivative's values are not calculated at the end of the frame time but at some intermediate point depending on the algorithm RK4 or RK2.

```
DYNAMIC

  DERIVATIVE NAME 1

    SPECIFY INDEPENDENT INTEGRATION ALGORITHM FOR
    NAME 1

    SPECIFY INDEPENDENT INTEGRATION STEP SIZE FOR
    NAME 1

      DIFFERENTIAL EQUATION OR TRANSFER FUNCTION
      DESCRIBING SYSTEM DYNAMICS INCLUDED IN NAME 1
      DERIVATIVE SECTION.

END $ 'OF NAME 1 REGION'

DERIVATIVE NAME 2

  SPECIFY INDEPENDENT INTEGRATION ALGORITHM FOR
  NAME 2

  SPECIFY INDEPENDENT INTEGRATION STEP SIZE FOR
  NAME 2

    DIFFERENTIAL EQUATIONS OR TRANSFER FUNCTIONS
    DESCRIBING SYSTEM DYNAMICS INCLUDED IN NAME 2
    DERIVATIVE SECTION.

END $ 'OF NAME 2 REGION'

                o
                o
                o

DERIVATIVE NAME N

                o
                o
                o

END $ 'END OF NAME N REGION'
```

Figure C-1. Basic multiple derivative configuration for ACSL.

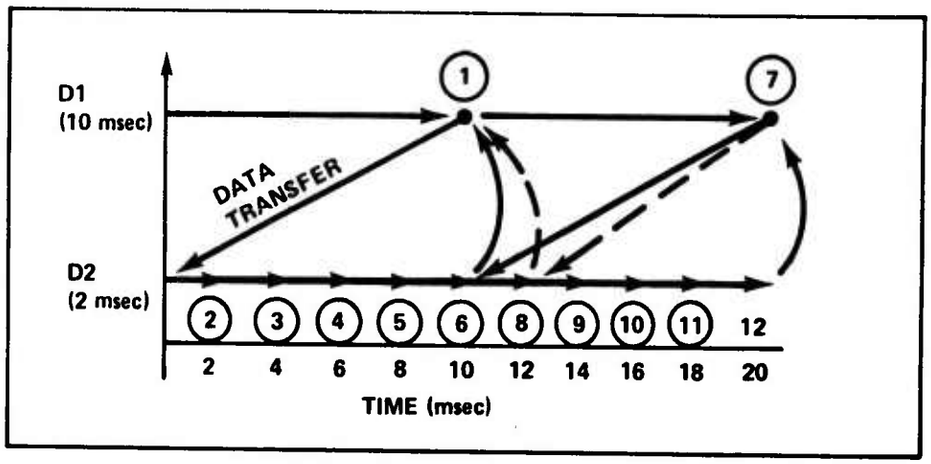


Figure C-2. Sequence of execution events.

The second major modification made on the ACSL system was to include hybrid drivers for ADC/DAC real time operation. The major motivation in this modification and development process is to use a simulation language to the maximum extent practical as the digital portion of the hybrid computer operation. This permits the user to make maximum use of the power, flexibility, and cost saving in using a higher level simulation language. As shown in Figure C-3, the analog functions, interface operation, and digital functions included in a hybrid computer are implemented using the ACSL in an all-digital environment with the functional areas associated with different derivative sections. The digital function could be further represented with multiple derivatives based on system dynamics. A significant cost saving in computer programming is realized when as much as practical of the digital function simulated by ACSL is used as the digital computer operation in the real time hybrid environment. This capability has been missing from previously available simulation languages; and considerable reprogramming was involved in the transition from an all-digital to a hybrid computer operation. An immediate observation is that it is not necessary for the total simulated hybrid computer to operate in real time but only the digital functions which will be used for real time hybrid computer operations. The analog function or hardware simulation can be replaced by the actual system hardware. In addition, the effects of various combinations of step sizes and integration algorithms can be investigated for the digital portion to be used in the hybrid simulation.

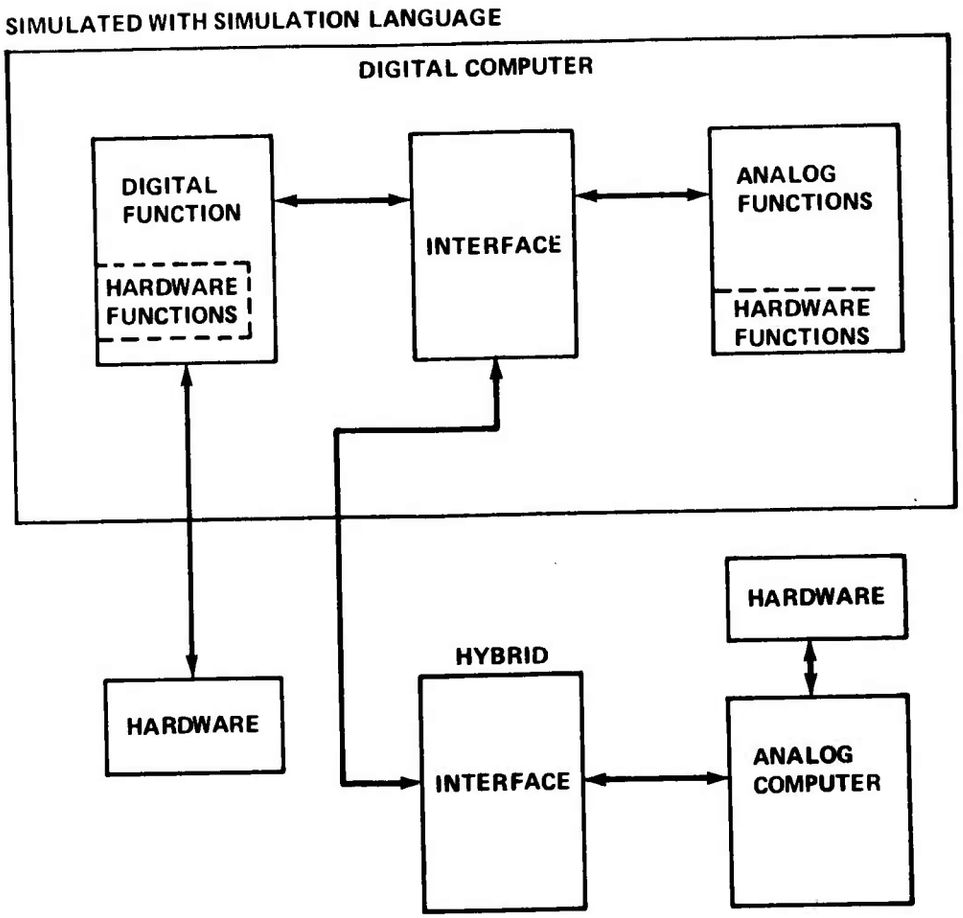


Figure C-3. Hybrid computer operation HWIL using ACSL.

2. Advantages of Using a Simulation Language

a. Digital Simulations

Developing a digital simulation using a procedural language such as FORTRAN requires the development of a number of special operators. These include overhead and bookkeeping functions, data input and output, formatting and manipulation, developing integration algorithms, plotting routines, etc. The programs must be structured so that all aspects of the design specification of the simulation can be met. FORTRAN is a procedural language and was not designed specifically for simulation application. The desired simulation structure typically cannot be achieved without special expertise. A typical example would be the effort involved to translate a second-order transfer function into a FORTRAN-coded program, e.g., identifying the first-order differential equations, initial conditions, developing integration

algorithms, formats for data input and output, plotting routines, and ensuring the proper sequence of program execution. The use of a simulation language significantly simplifies the task placed on the simulation developer. Another frequently overlooked advantage to simulation languages is that a standard procedure for documentation is readily available. The ACSL, as used in the ASC, includes a specifically developed post processor to assist the simulation user in performing statistical analysis.

A simulation produces raw data which must be presented in such a way as to be useful in the analysis of the simulated system. ACSL provides the OUTPUT, PRINT, PLOT, DISPLAY, and RANGE commands as a basic set of operations used to present simulation data. Simulation analysis may require presentation of more than just the raw data provided on a per case basis by a series of simulation runs. To meet the need for in depth analysis of simulation data, ASC has funded the development of a data processing program called the postprocessor.

The postprocessor accepts previously generated ACSL simulation data, permits arbitrary mathematical and statistical operations on the data, and presents the results in convenient form. All operations are specified by the user through standard commands (ADD, SQRT, AVERAGE, PLOT, etc.) read, and interpreted by the postprocessor. In addition, the user can easily add arbitrary operation to the standard commands to meet special needs.

The postprocessor is intended to replace the many special purpose programs that are written and rewritten with each simulation for the single purpose of presenting results in a useful form. Some postprocessor applications are summarized in the following paragraphs.

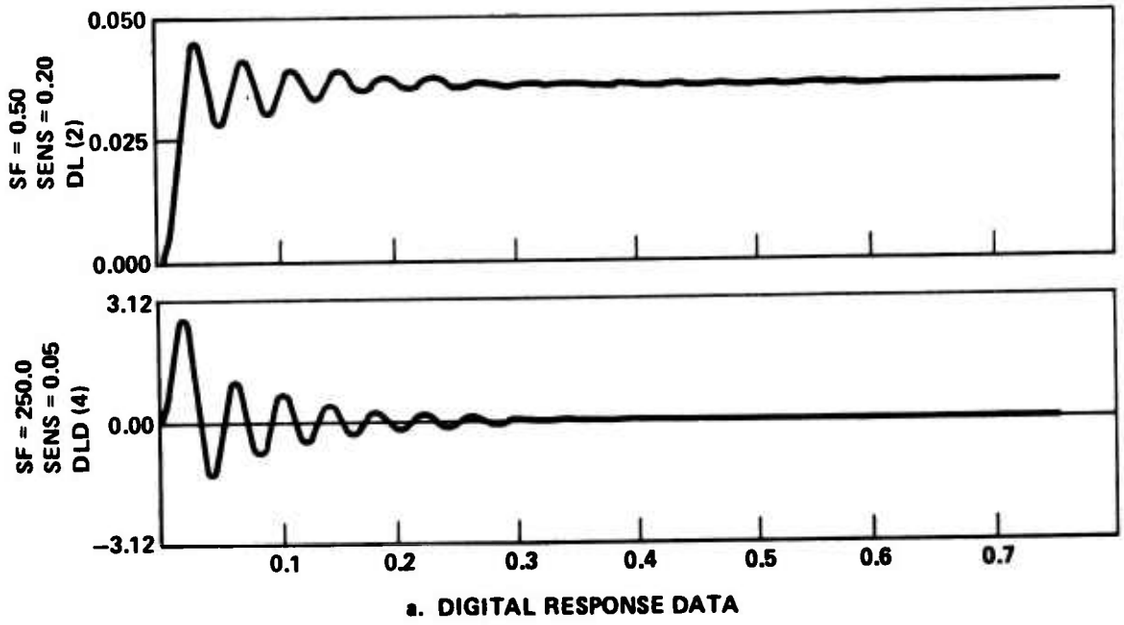
(1) Data Maintenance. It would be possible to make a series of ACSL simulation runs which produce much data but present only a summary of results. The great volume of raw data could be maintained on magnetic tape for later detailed analysis by the postprocessor, if or when required.

(2) Monte Carlo Analysis. ACSL simulations can be set up to iterate for a series of runs while randomly varying selected parameters to produce results suitable for Monte Carlo analysis. Any or all separate runs in a Monte Carlo sequence could be averaged together and plotted along with their corresponding standard deviations.

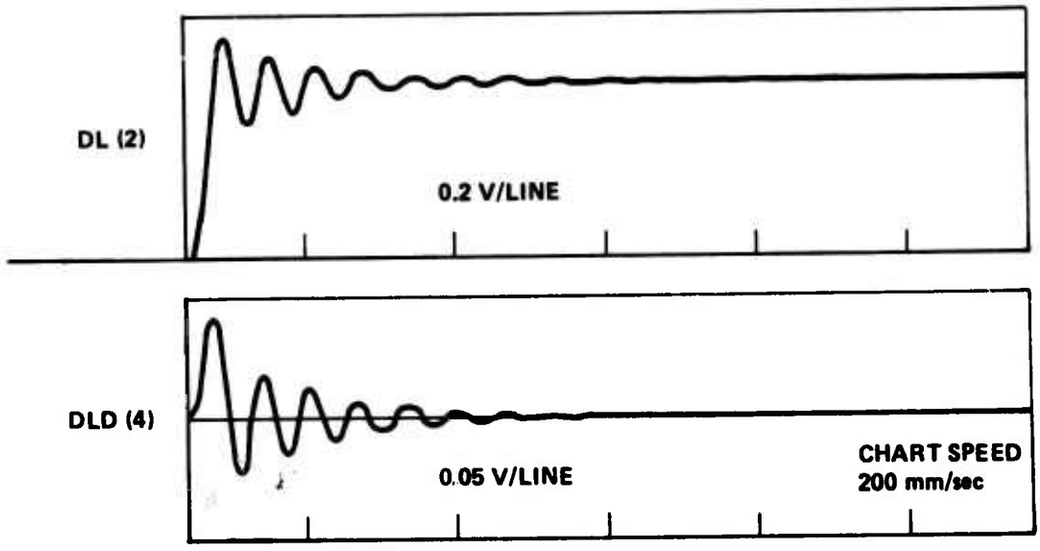
(3) Telemetry. Simulated test data can be generated and plotted as stripchart overlays for comparison with actual test telemetry at the test site. Later, digitized telemetry data can be easily input to the postprocessor and merged with simulated data for further comparison.

2. Support of Hybrid Computer Simulation

A digital computer simulation program using a flexible simulation language provides a check solution on hybrid computer results. This check solution capability is particularly important during the various stages of hybrid computer simulation development. Specifically, the nontrivial task of partitioning the mathematical model between the analog and digital computer can be studied for maximum utilization of resources, to investigate acceptable data sampling rates, to study the effects of frame-time compensation techniques, and to determine the appropriate integration algorithm for hybrid computer simulation. Using a standard digital plotter routine, a digital-generated scaled check solution of the analog computer can be obtained. A typical representation of comparative check solutions is shown in Figure C-4. The scaled digital results are used as overlays on the analog results for further verification of simulation implementation. Mathematical model changes and simulation updates are frequently required throughout the life of a simulation. Changes in the implementation of hybrid simulation are time consuming and subject to implementation errors; therefore, investigative changes are performed reluctantly. A digital simulation program that represents the hybrid computer simulation is a very effective method for investigating proposed changes in the simulation models. Only those changes that are determined to be essential are implemented in the hybrid simulation.



a. DIGITAL RESPONSE DATA



b. ANALOG RESPONSE DATA

Figure C-4. Typical comparative check solutions.

Appendix D. HYBRID IMPLEMENTATION PROCEDURES AT THE ASC

To maximize operational efficiency and minimize time, cost, and errors, implementation of hybrid simulations (including real time coding) is conducted according to a well-defined procedure (Addendum 1). A hybrid compiler (ECSSL), which runs in batch on the CDC 6600, is used as an aid in the implementation process. HOI software is used to enable automated setup and checkout of the analog math models. A standard documentation package (Appendix E) is developed to provide a permanent description of the implemented simulation so that changes which are inevitable can be accomplished in an orderly manner.

At the completion of the simulation development phase (which includes math modeling, ACSL programming, and model verification), it is presumed that the following activities have been completed:

1) The total math model package has been completed to the point that a high degree of confidence has been demonstrated by thorough verification. This math package, including listings, will be furnished.

2) Partitioning of the math models and simulation operations has been completed for each resource to be used.

3) Maximum values for all state variables have been determined and dynamic test run (from ACSL) plots have been produced to be used in analog and hybrid verification.

If the preceding steps have been accomplished, hybrid implementation proceeds according to the procedure established.

The ACSL source code for that part of the math model to be implemented on the analog computer is extracted, converted for syntax compatibility, and used as input to the hybrid compiler, ECSSL. ECSSL reduces, orders, and scales the equations, assigns components, develops connectional (patching) statements, calculates static check values, and prepares an HOI input tape. An auxiliary program creates coded analog symbols for use in constructing an analog diagram of the models for patching and other uses. (An updated version of the analog diagrammer to be completed in FY77 will create complete diagrams). ECSSL also provides trunk connections and MVFG patching. The HOI code is used to set up and checkout the analog processor(s), function generator, and the analog and discrete trunks required for the simulation interface. A static and dynamic check of each analog processor can be performed with HOI. The FORTRAN code from ACSL or from manual coding for the digital part of the program is augmented with real time code and compiled for the digital processor, analog/digital interface, direct digital-to-digital interfaces from central computer to cell computers, and control for the total simulator. ACSL has been upgraded by the addition of some of the real time operating software so that a portion of the augmentation derives directly from the ACSL translator.

Verification of the real time simulation proceeds from the subsystem, or module, level to the total simulation through the following steps:

1) Analog static check.

2) Digital-only debug.

3) Analog dynamic checks.

4) Hybrid dynamic check

5) Total simulation dynamic verification including cell and hardware.

The key to efficiency and accuracy in implementing and maintaining a hybrid or HWIL simulation is in the use and updating of the standard documentation package and the ACSL program. As changes in coding or patching are made, these are reflected in the documentation so that at any given time the documentation corresponds to the implemented simulation. This also provides a historical track of the development.

Addendum 1. HYBRID SIMULATION IMPLEMENTATION STANDARD OPERATING PROCEDURE

(10 June 1976)*

a) All hybrid models which can be handled by the hybrid compiler will be processed through the compiler, whether first time through or modification to existing simulation, unless a waiver is obtained from the Hybrid System Technology group leader prior to start of implementation.

b) Hybrid simulations will be permitted to begin patching analog patch boards only after the following items have been completed:

1) An error-free hybrid compiler run.

2) A tape or tapes of HOI source.

3) Error-free off-line HOI run.

4) Tape backup of all HOI files.

5) Volumes 1 and 2 of the standard simulation documentation package, including those items contained in the example in the enclosure. The software packages for automatic generation of documentation will be utilized for consistency, accuracy, and efficiency (analog diagrammer, TSU, SIMDOC, etc.).

6) Function files, if used.

7) Patching for CMVAFG, if used.

8) Schedule data entered for expected debug time.

9) Patch panel assignment for analog and logic.

10) Review and approval obtained to proceed from Hybrid Systems Group leader.

c) Once patch panels are patched, debugging with the analogs will begin during the scheduled times with all corrections being transferred to the standard simulation documentation referenced in 2e). Documentation update will normally occur each day of debugging but will always be complete at least once each week during scheduled debugging time. More time requires proper justification and approval by the Hybrid Systems Technology group leader.

d) Patch panels will be removed from the equipment and stored in the proper drawer at the end of each scheduled debug or production period.

---

*Hybrid Systems Technology Advanced Simulation Center

e) Prior to dynamic tests the following items must be completed:

1) The HOI files will be edited to include all information necessary for an individual who is familiar with HOI, but who is not familiar with the particular simulation, to completely set up and static check the simulation with no assistance from individuals familiar with the particular simulation.

2) Schedule data entered for dynamic test/debug phase.

3) Dynamic comparison data are collected, reviewed, and approved by the simulation team leader.

f) Prior to start of production the following items must be completed:

1) All modifications made during static and dynamic verification must be documented in the Standard Simulation Documentation package.

2) Production time scheduled.

3) Dynamic results reviewed and approved by simulation team leader.

g) All simulation modifications made after start of production will be documented in the Standard Simulation Documentation no later than 1 week after modifications are implemented.

Appendix E. JET ENGINE SIMULATION

JET ENGINE SIMULATION

TEAM LEADER-K.L.HALL
SIMULATION TEAM-TAL ADAMS,DICK MOHLERE,RICK KENDALL,JOHN VANSWEARINGEN

12 JANUARY 1977

US ARMY MISSILE COMMAND
GUIDANCE AND CONTROL DIRECTORATE
ADVANCED SIMULATION CENTER
REDSTONE ARSENAL ALABAMA 35809

CONTENTS

1.0 HOI LISTING

0945 11 JAN 1977 VERSION

```
1.001     "J65JET":
1.002     : 60.1;
1.010     "HUI EXECUTIVE FOR SETUP AND CHECK OF APSE PROGRAMS"
1.011     "VERSION KLH0112/6"
1.012
1.020     "OFF-LINE CHECK": @T, 0; @T, I;
1.030     @IX, @NA, XMODE;
1.040     NORMAL;
1.050     "PARAM & STATIC TEST VAL OF VARS": 11; 15; 16; 13;
1.060     "AUX PARAM": 48; 12;
1.070     "COEF": 21;
1.075     "LIM": 22;
1.076     "DCFG": 25;
1.078     "MVFG": 26;
1.080     "DERIV": 42;
1.090     "AMPL": 43; 44;
1.100     "FUNCTION GENERATORS": 45; 46;
1.110     "EXTERNAL TRUNKS": 47; 49;
1.120     "COMPARATORS": : 48;
1.122     :
1.130     "COEF OUTPUTS": 31;
1.140     .0001, VERIFY;
1.150     "DERIV CHK": 32;
1.160     "AMPL CHK": 33;  34;
1.170     "FUN GENS": 35; 36;
1.180     "EXT TRKS": 37; 39;
1.190     "CMP CHK": 38;
1.200     NORMAL;
1.210     "OFF-LINE CHECK COMPLETE":
1.220      "GO TO ON-LINE CHECK"H; 2.

2.010     "ROUTINE TO ALLOW SELECTION OF SETUP AND VERIFY OPTIONS"
2.020
2.030     @D, @T, I; GO=0, N=0, Y=1, N; @T, 0; @T, I;
2.040     : "SELECT DESIRED OPTION FROM THE FOLLOWING LIST AND ENTER THE NUMBER":
2.050     "FOR THE OPTION - DEFAULT IS ON-LINE SETUP AND CONNECTION CHECK": :
2.060     "     0 - ON-LINE SETUP AND CONNECTION CHECK":
2.070     "     1 - ON-LINE EQUATION CHECK":
2.080     "     2 - SETUP FOR PRODUCTION":
2.090     "     3 - RUN OFF-LINE CHECK":
2.100     "     4 - GO TO MONITOR":
2.110     "     5 - GO TO ANOTHER VIR MEM FILE":
2.120     "     6 - DUMP ALL PARTS TO MAG TAPE":
2.130     "     7 - DUMP ALL PARTS TO SOURCE FILE ON DISC 1":
2.140     "     8 - MAKE LISTING OF ALL PARTS AND VARIABLES ON LINE PRINTER":
2.150     "     9 - SET TEST DAC'S":
2.160     :
2.170     OPT=0, OPT←
2.180     OPT==0? $CLOSE: 3.
2.190     OPT==1? $CLOSE: 3.11.
2.200     OPT==2? $CLOSE: 9; 2.
2.210     OPT==3? $CLOSE: 1.
2.220     OPT==4? R;
2.230     OPT==5? : @HOID, MF; 2.
2.240     OPT==6? 60; 2.
2.250     OPT==7? "DELETE PRESENT COPY FROM DISC": GO← @TEMP, , @D1, U; $PAR: $END: $CLOSE: 2.
2.260     OPT==8? 50; 2.
```

```
2.270     OPT=-9? 17; 2.
2.999     : "***TRY AGAIN - FAILED TO MAKE A CHOICE*****": : 2.

3.010     "ON-LINE SETUP AND CHECKOUT":
3.020     @IX, @WA, XMODE;
3.030     NORMAL;
3.040     "PARAM & STATIC TEST VAL OF VARS": 11; 15; 16; 13;
3.050     "AUX PARAM": 48; 12;
3.060     "IS ANALOG COMPUTER ON-LINE? ": GO←
3.070     681, C; CON, CON, CON, CON, CON, CON, U; 1, C; 17;
3.072     : "SET POTS? (Y OR N,  DEFAULT IS N)": SP=0
3.074     SP← SP==0? 3.09.
3.080     "SET COEF, LIM &FUN": @SP, @SD, M; 21; 22; 25; 26;
3.090     "ON-LINE CHECK": 4;
3.091
3.092     80; "PRINT SPECIAL NOTES"
3.093
3.100     "ON-LINE CHECK COMPLETE"2.
3.110     "ON-LINE CHECK AGAINST EQUATIONS":
3.112     3.02; 3.03; 3.04; 3.05; 3.06; 3.07; 5;
3.120     : "EQUATION CHECK COMPLETE": 80; 2.

4.010     @SP, @SD, M; @VL, XMODE;
4.020     "COEF CHK": .0005, V; 21; 22;
4.030     "CHECK CONNECTIONS": :
4.040     @ST, M; .001, V; @RU, M; @SD, M;
4.050     "DERIV CHK": 32;
4.060     "AMPL CHK": 33; 34;
4.070     "FUN GEN CHK": 35; 36;
4.080     "EXT TRK CHK": 37; 39;
4.090     "CMP CHK": 38;
4.100     NORMAL;

5.010     M; .001, V; @VL, XMODE;
5.020     @ST, M; "DERIVATIVES": 42;
5.030     "AMP CHK": 43; 44;
5.040     "FUN GEN CHK": 46;
5.050     "EXT TRK CHK": 47; 49;
5.060     "CMP CHK": 48;
5.070     NORMAL;

9.001     : "SETUP FOR PRODUCTION": :
9.003     "CHECKING TRUNKING STATION": :← M;
9.010     "  CONECTORS":
9.020     @IX, @WA, X; FLAG=0
9.030     @AD, X;
9.035     "THE FOLLOWING CONNECTOR EXAMPLES MUST BE MODIFIED FOR EACH SIMULATION"
9.050     TKP1=9W02, TKP1<>@001? FLAG=1, : "6600 ADC'S - CON W02 TO V01":
9.400     FLAG<>0? : "PATCH T/S OR ENTER 1 FOR GO TO BYPASS T/S PATCH CHECK":
    : GO← GO==1? GO=0, FLAG=0, 9.61.
9.410     FLAG<>0? 9.
9.420     : "T/S PATCHING ***OK*** FOR PRODUCTION": : FLAG=0
9.610     @SP, @SD, M; N;
9.620     : "INPUT PRODUCTION BETA VALUE (DEFAULT IS BETA = 1)": : BETA=1, BETA←
9.625     BETA==1? 9.7.
9.630     : "SETTING POTS AFTER BETA CHANGE": : 21; 22;
9.640     : "VERIFYING POTS AFTER BETA CHANGE": : .0005, V; 21; 22;
9.650     : "POTS VERIFIED FOR BETA CHANGE": : N;
9.700     : : "SETTING POT CHANGES FOR PRODUCTION": 10;
9.710     : "VERIFYING PRODUCTION POT CHANGES": .0005, V; 10; :
9.720     : : : "PRODUCTION SETUP AND CHECK COMPLETE": :
```

```
 9.730    N; @IC, @SD, M;

10.001    "PRODUCTION POT LIST"

11.001    "INPUT PARAMETER VALUES AND PARAMETRIC SCALINGS"
11.002    BETA=1.00000
11.003    N=0, Y=1, CON=1, GO=0
11.101    NCO=8.00000*10**3
11.102    WFO=9.14400*10**(-1)
11.103    NCOM=0.00000
11.104    K1=1.20000
11.105    K2=2.00000
11.201    ↑TIME=5.00000*10**1
11.202    ↑WC=2.00000*10**2
11.203    ↑N=1.00000*10**4
11.204    ↑NEI=1.00000*10**4
11.205    ↑NE=1.00000*10**4
11.301    ↑AAV001=1.00000*10**1
11.302    ↑AAV002=4.00000
11.303    ↑P3=3.00000*10**2
11.304    ↑AAV003=1.08500*10**2
11.305    ↑WCF3=1.40000*10**2
11.306    ↑AAV004=1.48700
11.307    ↑AAV005=7.00000*10**(-1)
11.308    ↑T3=1.50000*10**3
11.309    ↑AAV006=1.07300*10**2
11.310    ↑WCF6=1.40000*10**2
11.311    ↑DH3=3.00000*10**2
11.312    ↑P4=3.00000*10**2
11.313    ↑AAV008=9.90000*10**(-1)
11.314    ↑AAV009=5.94000
11.315    ↑AAV010=2.13142*10**4
11.316    ↑T4=3.00000*10**3
11.317    ↑AAV011=2.89000*10**1
11.318    ↑AAV013=8.67000*10**3
11.319    ↑WCLP=2.00000*10**2
11.320    ↑AAV014=2.50000*10**(-1)
11.321    ↑AAV015=1.17760
11.322    ↑AAV016=2.94400*10**(-1)
11.323    ↑DENT=3.00000
11.324    ↑T5=3.00000*10**3
11.325    ↑DH4=3.00000*10**2
11.326    ↑NC=1.00000*10**4
11.327    ↑DH43=3.00000*10**2
11.328    ↑AAV019=6.00000*10**4
11.329    ↑AAV020=8.09315*10**4
11.330    ↑DWF=1.00000*10**1
11.331    ↑WF=6.00000
11.332    ↑NECS=1.00000*10**4
11.333    ↑P4N=1.00000*10**1

12.001    "AUXILIARY PARAMETRIC SCALINGS"
12.101    ↑AAV007=(↑T3)**2
12.102    ↑AAV012=SQR(↑T4)
12.103    ↑AAV017=(↑T4)**2
12.104    ↑AAV018=(↑T5)**2

13.001    "ENTER MVFG PHYSICAL ASSIGNMENT DEFINITION INTO MVGDIR"

15.001    "STATIC TEST VALUES OF DIFFERENTIAL VARIABLES"
15.101    TIME=1.00000*10**1
```

```
15.102      WC=9.36600*10**1
15.103      N=7.00000*10**3
15.104      NEI=1.00000*10**3
15.105      NE=1.00000*10**3
15.501      F1[! ]=(1., 9.7*10**-1, 1.2, 1.34, 1.54, 1.74, 2.03, 2.39, 2.93, 3.76, 5.28, 6.46, 7.
      33, 8.23, 9.16, 1.*10**1)
15.502      DUMN[! ]=(0.0, 1.5*10**3, 3.*10**3, 3.5*10**3, 4.*10**3, 4.5*10**3, 5.*10**3, 5.
      5*10**3, 6.*10**3, 6.5*10**3, 7.*10**3, 7.5*10**3, 8.*10**3, 8.3*10**3, 8.6*10**3, 9.5*
      10**3)
15.503      F3[! ]=(0.0, -4.5, -1.4, 2.4, 6.7, 1.27*10**1, 2.04*10**1, 2.9*10**1, 4.*10**1, 5.
      25*10**1, 6.62*10**1, 8.01*10**1, 9.1*10**1, 9.75*10**1, 1.025*10**2, 1.065*10**2)
15.504      F4[! ]=(0.0, 2.04*10**-1, 3.63*10**-1, 4.05*10**-1, 4.69*10**-1, 5.1*10**-1, 5.
      33*10**-1, 6.23*10**-1, 6.99*10**-1, 7.84*10**-1, 9.27*10**-1, 1.03, 1.103, 1.218, 1.33
      9, 1.487)
15.505      F6[! ]=(0.0, -7.15, -5., -1., 3.5, 1.05*10**1, 1.85*10**1, 2.75*10**1, 3.825*10**
      1, 5.12*10**1, 6.53*10**1, 7.92*10**1, 9.05*10**1, 9.63*10**1, 1.012*10**2, 1.073*10**
      2)
15.506      F2[! ]=(0.0, 7.*10**-2, 1.15*10**-1, 1.15*10**-1, 9.*10**-2, 8.*10**-2, 5.*10**
      -2, 0.0, -8.*10**-2, -1.9*10**-1, -3.4*10**-1, -5.8*10**-1, -1., -2.5, -3.25, -4.)
15.507      DMWCF3[! ]=(0.0, 5., 1.*10**1, 1.5*10**1, 2.*10**1, 2.1*10**1, 2.2*10**1, 2.3*10
      **1, 2.4*10**1, 2.5*10**1, 2.6*10**1, 2.7*10**1, 2.8*10**1, 2.9*10**1, 2.95*10**1, 3.*1
      0**1)
15.508      F5[! ]=(0.0, -3.4*10**-1, -5.5*10**-1, -1.*10**-1, -1.42*10**-1, -1.82*10**-1,
      -1.97*10**-1, -2.18*10**-1, -2.3*10**-1, -2.43*10**-1, -2.61*10**-1, -2.83*10**-1, -3
      .12*10**-1, -3.4*10**-1, -5.3*10**-1, -7.*10**-1)
15.509      DMWCF6[! ]=(0.0, 3., 5., 1.*10**1, 1.5*10**1, 2.*10**1, 2.2*10**1, 2.4*10**1, 2.5
      *10**1, 2.6*10**1, 2.7*10**1, 2.8*10**1, 2.9*10**1, 2.95*10**1, 3.*10**1, 3.02*10**1)
15.510      F7[! ]=(4.6*10**-1, 5.7*10**-1, 6.5*10**-1, 7.07*10**-1, 7.9*10**-1, 8.2*10**-
      1, 8.67*10**-1, 9.*10**-1, 9.24*10**-1, 9.35*10**-1, 9.666*10**-1, 9.84*10**-1, 9.9*10
      **-1, 9.9*10**-1, 9.9*10**-1, 9.9*10**-1)
15.511      DMP4[! ]=(2., 3., 4., 5., 7., 8., 1.*10**1, 1.2*10**1, 1.4*10**1, 1.5*10**1, 2.*10**
      1, 2.5*10**1, 3.*10**1, 1.*10**2, 2.*10**2, 3.*10**2)
15.512      F8[! ]=(1.13*10**1, 1.54*10**1, 1.96*10**1, 2.22*10**1, 2.4*10**1, 2.54*10**1,
      2.63*10**1, 2.7*10**1, 2.75*10**1, 2.818*10**1, 2.839*10**1, 2.852*10**1, 2.871*10**1
      , 2.882*10**1, 2.888*10**1, 2.89*10**1)
15.513      DMP4P2[! ]=(1.061, 1.117, 1.236, 1.363, 1.488, 1.617, 1.754, 1.903, 2.062, 2.416, 2
      .599, 2.843, 3.329, 3.995, 5.146, 9.71)
15.514      F9[! ]=(9.1*10**-3, 1.83*10**-2, 3.57*10**-2, 5.22*10**-2, 6.81*10**-2, 8.31*1
      0**-2, 9.76*10**-2, 1.116*10**-1, 1.258*10**-1, 1.499*10**-1, 1.617*10**-1, 1.731*10**
      -1, 1.966*10**-1, 2.146*10**-1, 2.388*10**-1, 2.5*10**-1)

16.001      "STATIC TEST VALUES OF ALGEBRAIC VARIABLES"
16.101      DER001=1.00000"DER(TIME)"
16.102      AAV001=5.28000
16.103      AAV002=-7.73200*10**(-1)
16.104      P3=1.34844*10**2
16.105      AAV003=6.62000*10**1
16.106      WCF3=2.74600*10**1
16.107      AAV004=9.27000*10**(-1)
16.108      AAV005=-2.93440*10**(-1)
16.109      T3=8.47818*10**2
16.110      AAV006=6.53000*10**1
16.111      WCF6=2.83600*10**1
16.112      AAV007=7.18795*10**5
16.113      DH3=8.01873*10**1
16.114      P4=1.28102*10**2
16.115      AAV008=9.90000*10**(-1)
16.116      AAV009=1.22206
16.117      AAV010=9.36371*10**2
16.118      T4=1.78419*10**3
16.119      AAV011=2.88350*10**1
```

```
16.120      AAV012=4.22397*10**1
16.121      AAV013=3.69380*10**3
16.122      WCLP=8.74485*10**1
16.123      DER002=-6.21151*10**3"DER(WC)"
16.124      AAV014=2.19135*10**(-1)
16.125      AAV015=1.02556
16.126      AAV016=2.24734*10**(-1)
16.127      DENT=1.22474
16.128      T5=1.45680*10**3
16.129      AAV017=3.18333*10**6
16.130      AAV018=2.12226*10**6
16.131      DH4=8.76352*10**1
16.132      NC=8.00000*10**3
16.133      DH43=7.44793
16.134      AAV019=6.97573*10**2
16.135      AAV020=4.70463*10**2
16.136      DER003=4.70463*10**2"DER(N)"
16.137      DWF=3.20000*10**(-1)
16.138      WF=1.23440
16.139      NECS=5.00000*10**2
16.140      DER004=1.00000*10**3"DER(NEI)"
16.141      DER005=-5.00000*10**3"DER(NE)"
16.142      P4N=4.28146

17.002      : "CHECKING TEST DAC CONNECTIONS": :
17.004      @AD; X; @IX; @WA; XMODE; FLAG=0
17.006      TVP1=9W41; TVP1<>@X02? FLAG=1; : "TEST MDAC INPUT - CON W41 TO V02":
17.400      FLAG<>0? : "PATCH T/S OR ENTER 1 FOR GO TO BYPASS T/S PATCH CHECK":
     : GO← GO==1? GO=0; FLAG=0; 17.5.
17.410      FLAG<>0? 17.
17.420      : "T/S PATCHING ***OK*** FOR STATIC TEST": : FLAG=0
17.500      1DA00=0
17.510      1DA01=0
17.520      1DA02=0
17.530      1DA03=0
17.540      1DA04=0
17.550      1DA05=0
17.560      1DA06=0
17.570      1DA07=0
17.580      1DA08=0
17.590      1DA09=0
17.600      1DA10=0
17.610      1DA11=0
17.620      1DA12=0
17.630      1DA13=0
17.640      1DA14=0
17.650      1DA15=0
17.999      : "DAC TEST VOLTAGES SET": :

21.001      1C001=+↑AAV003/↑WCF3
21.002      1C002=+1.00000/(↑TIME*1*BETA)
21.003      1C003=+2.99200*10**1*↑AAV001/↑P3
21.004      1C004=+NEI/↑NEI
21.005      1C005=+9.50000*10**(-1)*↑P3/↑P4
21.006      1C006=+↑AAV006/↑WCF6
21.007      1C007=+2.99200*10**1*↑AAV002/↑P3
21.008      1C008=+1.00000*10**(-1)*↑WC/↑WCF6
21.010      1C010=+2.26500*10**(-2)*↑T3/1DH3
21.011      1C011=+5.19000*10**2*↑AAV004/↑T3
21.012      1C012=+↑T3/↑T4
21.013      1C013=+5.19000*10**2*↑AAV005/↑T3
```

```
21.015      1C015=+5.19000*10**2/↑T3
21.016      1C016=+1.20976*10**2/↑DH3
21.017      1C017=+1.27050*10**(-5)*↑AAV007/↑DH3
21.018      1C018=+(NCD+NCDM)*1.00000*10**(-1)/↑NC
21.020      1C020=+1.00000*10**(-1)*↑DWF/↑WF
21.021      1C021=+K2*1.00000*10**(-4)*↑NEI/↑DWF
21.022      1C022=+K1*1.00000*10**(-4)*↑NE/↑DWF
21.023      1C023=+WFD/↑WF
21.025      1C025=+1.18400*10**(-4)*↑T4/↑AAV015
21.026      1C026=+↑NECS/(↑NE*1*BETA)
21.027      1C027=+↑T4/(↑T5*↑DENT)
21.028      1C028=+↑AAV009/(↑WF*↑AAV008)
21.030      1C030=+7.17647*10**4*↑AAV009/(↑AAV010*↑WC)
21.031      1C031=+1.23680*10**(-1)/↑AAV015
21.032      1C032=+1.00000*10**(-1)*↑AAV010/↑T4
21.033      1C033=+↑NC/↑NECS
21.035      1C035=+NE/↑NE
21.036      1C036=+↑N/↑NECS
21.037      1C037=+↑NE/(↑NEI*1*BETA)
21.038      1C038=+↑NE/(↑NE*1*BETA)
21.040      1C040=+WC/↑WC
21.041      1C041=+3.34225*10**(-3)*↑P4/↑P4N
21.042      1C042=+1.00000*10**(-1)*↑WC/↑WCF3
21.043      1C043=+↑AAV013/(↑P4*↑AAV011)
21.045      1C045=+↑AAV013/(↑WCLP*↑AAV012)
21.046      1C046=+1.00000*10**3*↑WC/(↑WC*1000*BETA)
21.047      1C047=+1.00000*10**3*↑WCLP/(↑WC*1000*BETA)
21.048      1C048=+↑AAV016/(↑AAV014*↑AAV015)
21.050      1C050=+N/↑N
21.051      1C051=+1.00000/↑DENT
21.052      1C052=+↑AAV016/↑DENT
21.053      1C053=+2.26500*10**(-2)*↑T4/↑DH4
21.055      1C055=+2.26500*10**(-2)*↑T5/↑DH4
21.056      1C056=+1.27050*10**(-5)*↑AAV018/↑DH4
21.057      1C057=+1.27050*10**(-5)*↑AAV017/↑DH4
21.058      1C058=+↑DH3/↑DH43
21.060      1C060=+↑DH4/↑DH43
21.061      1C061=+4.72100*10**3*↑AAV019/(↑AAV020*↑N)
21.062      1C062=+↑AAV019/(↑DH43*↑WC)
21.063      1C063=+1.00000*10**(-1)*↑AAV020/(↑N*1*BETA)
21.999      1C000=+TIME/↑TIME

22.001      L001=5.00000*10**2/↑NECS, "LOWER"
            U001=5.00000*10**2/↑NECS, "UPPER"
            L001, U001: H; "1H001"

23.001      "A AMP VALUES "

24.001      "I AMP VALUES"

25.001      1G001(DUMN[! ]/(↑N))=(F3[! ]/(↑AAV003))"AAV003, N"
25.002      1G002(DUMN[! ]/(↑N))=(F4[! ]/(↑AAV004))"AAV004, N"
25.003      1G003(DUMN[! ]/(↑N))=(F6[! ]/(↑AAV006))"AAV006, N"
25.004      1G004(DMWCF3[! ]/(↑WCF3))=(F2[! ]/(↑AAV002))"AAV002, WCF3"
25.005      1G005(DMWCF6[! ]/(↑WCF6))=(F5[! ]/(↑AAV005))"AAV005, WCF6"
25.006      1G006(DMP4[! ]/(↑P4))=(F7[! ]/(↑AAV008))"AAV008, P4"
25.007      1G007(DMP4P2[! ]/(↑P4N))=(F8[! ]/(↑AAV011))"AAV011, P4N"
25.010      1G010(DMP4P2[! ]/(↑P4N))=(F9[! ]/(↑AAV014))"AAV014, P4N    @11JAN77TJA"
25.999      1G000(DUMN[! ]/(↑N))=(F1[! ]/(↑AAV001))"AAV001, N"

26.001      "FUN GEN & LOAD FUN DATA TO MVFG"
```

```
27.001    "EXTERNAL TRUNK VALUES"

28.001    "COMP VALUES"

31.001    1P001=+1C001*1F001
31.002    1P002=+1C002*(-1)
31.003    1P003=+1C003*1F000
31.004    1P004=+1C004*(-1)
31.005    1P005=+1C005*1A005
31.006    1P006=+1C006*1F003
31.007    1P007=+1C007*1F004
31.008    1P008=+1C008*1A040
31.010    1P010=+1C010*1A010
31.011    1P011=+1C011*1F002
31.012    1P012=+1C012*1A010
31.013    1P013=+1C013*1F005
31.015    1P015=+1C015*(-1)
31.016    1P016=+1C016*(-1)
31.017    1P017=+1C017*1A003
31.018    1P018=+1C018*(+1)
31.020    1P020=+1C020*1A020
31.021    1P021=+1C021*1A002
31.022    1P022=+1C022*1A034
31.023    1P023=+1C023*(-1)
31.025    1P025=+1C025*1A025
31.026    1P026=+1C026*1A026
31.027    1P027=+1C027*1A025
31.030    1P030=+1C030*1A008
31.031    1P031=+1C031*(+1)
31.032    1P032=+1C032*1A033
31.033    1P033=+1C033*1A016
31.035    1P035=+1C035*(+1)
31.036    1P036=+1C036*1A050
31.037    1P037=+1C037*1A035
31.038    1P038=+1C038*1A035
31.040    1P040=+1C040*(+1)
31.041    1P041=+1C041*1A004
31.042    1P042=+1C042*1A040
31.045    1P045=+1C045*1A018
31.046    1P046=+1C046*1A040
31.047    1P047=+1C047*1A043
31.050    1P050=+1C050*(-1)
31.051    1P051=+1C051*(-1)
31.052    1P052=+1C052*1A048
31.053    1P053=+1C053*1A025
31.055    1P055=+1C055*1A023
31.056    1P056=+1C056*1A028
31.057    1P057=+1C057*1A013
31.058    1P058=+1C058*1A011
31.060    1P060=+1C060*1A041
31.061    1P061=+1C061*1A053
31.063    1P063=+1C063*1A058
31.999    1P000=+1C000*(-1)

32.001    "DERIV CONNECTIONS"
32.002    1D002=+(1C037*1A035)
32.035    1D035=+(10*1C026*1A026+10*1C038*1A035)
32.040    1D040=+(1C047*1A043+1C046*1A040)"NMS CAP"
32.050    1D050=+(10*1C063*1A058)
32.999    1D000=+(1C002*(-1))
```

```
32.001    1A001=-(1C001*1F001+10*1C042*1A040)
33.002    1A002=-(1C004*(-1))"IC"
33.003    1A003=+(1A010**2)
33.004    1A004=-1A015
33.005    1A005=-(1C003*1F000+1C007*1F004)
33.006    1A006=-(1C006*1F003+10*1C008*1A040)
33.008    1A008=-(+1A021*1F006)/1C028
33.009    1A009=-1A038
33.010    1A010=-(1C011*1F002+1C013*1F005+1C015*(-1))
33.011    1A011=-(10*1C010*1A010+1C017*1A003+1C016*(-1))
33.013    1A013=-(1A025**2)
33.015    1A015=-(1C005*1A005)
33.016    1A016=-(10*1C018*(+1))
33.018    1A018=-(+1A015*1F007)/1C043
33.020    1A020=-(1C022*1A034+1C021*1A002)
33.021    1A021=-(10*1C020*1A020+1C023*(-1))
33.023    1A023=-(1C027*1A025)/1A036
33.025    1A025=-(1C012*1A010+10*1C032*1A033)
33.026    1A026=LIM(L001*(-1), U001*(+1), -(1C033*1A016+1C036*1A050))"1H001"
33.028    1A028=+(1A023**2)
33.030    1A030=-(1C025*1A025+10*1C031*(+1))
33.031    1A031=-(10*1C041*1A004)
33.033    1A033=-(1C030*1A008)/(-1A040)
33.034    1A034=-1A035
33.035    1A035=-(1C035*(+1))"IC"
33.036    1A036=-(1C052*1A048+1C051*(-1))
33.038    1A038=+SQR(-1A025)
33.040    1A040=-(1C040*(+1))"IC"
33.041    1A041=-(10*1C053*1A025+10*1C055*1A023+1C057*1A013+1C056*1A028)
33.043    1A043=-(1C045*1A018)/1A038
33.045    1A045=-(1C058*1A011+1C060*1A041)
33.048    1A048=-(-1F010*1A030)/1C048    "@11JAN77TJA"
33.050    1A050=-(1C050*(-1))"IC"
33.053    1A053=-(-1A045*1A040)/1C062
33.058    1A058=-(1C061*1A053)/1A050
33.999    1A000=-(1C000*(-1))"IC"

34.001    "I AMP CONNECTIONS"

35.001    1F001=1G001(1A050)
35.002    1F002=-1G002(1A050)
35.003    1F003=1G003(1A050)
35.004    1F004=1G004(1A001)
35.005    1F005=-1G005(1A006)
35.006    1F006=1G006(1A015)
35.007    1F007=1G007(1A031)
35.010    1F010=1G010(1A031),   "11JAN77TJA"
35.999    1F000=1G000(1A050)

36.001    "FUN GEN CONNECTIONS"

37.001    "EXT TRUNK CONNECTIONS"

38.001    "COMP CONNECTIONS"

39.001    "NONREADABLE EXT TRUNK CONNECTIONS"

42.001    "DERIV VALUES"
42.002    1D002=-DER004/(↑NEI*1*BETA)
42.035    1D035=+DER005/(↑NE*1*BETA)
```

```
42.040      1D040=+DER002/(↑WC*1000*BETA)
42.050      1D050=-DER003/(↑N*1*BETA)
42.999      1D000=-DER001/(↑TIME*1*BETA)

43.001      1A001=+(WCF3/↑WCF3)
43.002      1A002=+(NEI/↑NEI)
43.003      1A003=+(AAV007/↑AAV007)
43.004      1A004=-(P4/↑P4)
43.005      1A005=-(P3/↑P3)
43.006      1A006=+(WCF6/↑WCF6)
43.008      1A008=-(AAV009/↑AAV009)
43.009      1A009=-(AAV012/↑AAV012)
43.010      1A010=+(T3/↑T3)
43.011      1A011=-(DH3/↑DH3)
43.013      1A013=-(AAV017/↑AAV017)
43.015      1A015=+(P4/↑P4)
43.016      1A016=-(NC/↑NC)
43.018      1A018=-(AAV013/↑AAV013)
43.020      1A020=-(DWF/↑DWF)
43.021      1A021=+(WF/↑WF)
43.023      1A023=+(T5/↑T5)
43.025      1A025=-(T4/↑T4)
43.026      1A026=+(NECS/↑NECS)
43.028      1A028=+(AAV018/↑AAV018)
43.030      1A030=-(AAV015/↑AAV015)
43.031      1A031=+(P4N/↑P4N)
43.033      1A033=+(AAV010/↑AAV010)
43.034      1A034=+(NE/↑NE)
43.035      1A035=-(NE/↑NE)
43.036      1A036=+(DENT/↑DENT)
43.038      1A038=+(AAV012/↑AAV012)
43.040      1A040=-(WC/↑WC)
43.041      1A041=+(DH4/↑DH4)
43.043      1A043=+(WCLP/↑WCLP)
43.045      1A045=-(DH43/↑DH43)
43.048      1A048=-(AAV016/↑AAV016)
43.050      1A050=+(N/↑N)
43.053      1A053=+(AAV019/↑AAV019)
43.058      1A058=-(AAV020/↑AAV020)
43.999      1A000=+(TIME/↑TIME)

44.001      "I AMP VALUES"

45.001      1F001=+(AAV003/↑AAV003)
45.002      1F002=-(AAV004/↑AAV004)
45.003      1F003=+(AAV006/↑AAV006)
45.004      1F004=+(AAV002/↑AAV002)
45.005      1F005=-(AAV005/↑AAV005)
45.006      1F006=+(AAV008/↑AAV008)
45.007      1F007=+(AAV011/↑AAV011)
45.010      1F010=+(AAV014/↑AAV014)  "@11JAN77TJA"
45.999      1F000=+(AAV001/↑AAV001)

46.001      "MVFG VALUES"

47.001      "EXTERNAL TRUNK VALUES"

48.001      "COMP VALUES"

49.001      "NONREADABLE EXT TRUNK VALUES"
```

```
50.002      @T, 0; : "ROUTINE TO PRINT ALL NUMBERED PARTS ON THE LINE PRINTER": :
50.010      "MAKE LINE PRINTER READY AND TYPE RETURN": : GO←
50.020      @L, 0; : 60.1; : : $PAR: $VAR: @T, 0;

60.002      : : 60.2; "                VIRTUAL MEMORY DUMP TO TAPE": 60.2; : :
60.010      "TYPE 60.1; TO CHANGE DATE OR TIME,  THEN TYPE 60.2.": : :
60.100      "                    0945 11 JAN 1977  VERSION":
60.120      H;
60.200      "*************************************************************":
60.210      "WILL WRITE OVER ANY TAPE ON BOTTOM TAPE UNIT WITH WRITE ENABLED":
60.220      60.2; : : "MOUNT SCRATCH TAPE ON BOTTON UNIT AND GET READY": :
60.230      GO←
60.240      @M1, 0; $CLOSE: @M1, 0; $CLOSE:
60.250      @TEMP, , @M1, 0; $PAR: $END: $CLOSE:
60.260      : : 60.2; "                          FINISHED DUMPING": 60.2;
60.270      : : "TO ERASE VIRTUAL MEMORY AND RELOAD FROM SCRATCH ON BOTTON UNIT":
60.280      "TYPE $ALL\  AND THEN TYPE @TEMP, , @M1, I; ": :
60.900      "-------------------------------------------------------------":

80.001      : "#################### SPECIAL NOTES ####################": : :
80.999      60.9; :
```

```
OPT =  8.00000
1P061 =  .004069
1P060 =  .292117
1P058 = -.267291
1P057 = -.134814
1P056 =  .089877
1P055 =  .109988
1P053 = -.134706
1P052 = -.074911
1P051 = -.333333
1P050 = -.700000
1P047 =  .437242
1P046 = -.468300
1P045 = -.337196
1P042 = -.066900
1P041 = -.042814
1P040 =  .468300
1P038 = -.100000
1P037 = -.100000
1P036 =  .700000
1P035 =  .100000
1P033 = -.800000
1P032 =  .031212
1P031 =  .105027
1P030 = -.020573
1P027 = -.198243
1P026 =  .050000
1P025 = -.179389
1P023 = -.152400
1P022 =  .012000
1P021 =  .020000
1P020 = -.005333
1P018 =  .080000
1P017 =  .030441
1P016 = -.403253
1P015 = -.346000
1P013 =  .101530
1P012 =  .282606
```

```
1P011 = -.320742
1P010 =  .064010
1P008 = -.066900
1P007 = -.077113
1P006 =  .466429
1P005 = -.427006
1P004 = -.100000
1P003 =  .526592
1P002 = -.020000
1P001 =  .472857
1F000 =  .528000
1F010 =  .876540
1F006 =  1.00000
1F005 =  .419200
1F004 = -.193300
1F003 =  .608574
1F002 = -.623403
1F001 =  .610138
1A000 =  .200000
1A058 = -.005813
1A053 =  .011626
1A050 =  .700000
1A048 = -.763363
1A045 = -.024826
1A043 =  .437242
1A041 =  .292117
1A040 = -.468300
1A038 =  .771188
1A036 =  .408247
1A035 = -.100000
1A034 =  .100000
1A033 =  .043931
1A031 =  .428146
1A030 = -.870890
1A028 =  .235807
1A026 =  .050000
1A025 = -.594730
1A023 =  .485600
1A021 =  .205733
1A020 = -.032000
1A018 = -.426044
1A016 = -.800000
1A015 =  .427007
1A013 = -.353704
1A011 = -.267201
1A010 =  .565212
1A009 = -.771188
1A008 = -.205734
1A006 =  .202571
1A005 = -.449480
1A004 = -.427007
1A003 =  .319465
1A002 =  .100000
1A001 =  .196143
1D000 = -.020000
1D050 = -.047046
1D040 = -.031057
1D035 = -.500000
1D002 = -.100000
1G000[32] =  1.00000
1G000[31] =  .950000
```

```
1G000[30] =  .916000
1G000[29] =  .860000
1G000[28] =  .823000
1G000[27] =  .830000
1G000[26] =  .733000
1G000[25] =  .800000
1G000[24] =  .646000
1G000[23] =  .750000
1G000[22] =  .528000
1G000[21] =  .700000
1G000[20] =  .376000
1G000[19] =  .650000
1G000[18] =  .293000
1G000[17] =  .600000
1G000[16] =  .239000
1G000[15] =  .550000
1G000[14] =  .203000
1G000[13] =  .500000
1G000[12] =  .174000
1G000[11] =  .450000
1G000[10] =  .154000
1G000[9] =  .400000
1G000[8] =  .134000
1G000[7] =  .350000
1G000[6] =  .120000
1G000[5] =  .300000
1G000[4] =  .097000
1G000[3] =  .150000
1G000[2] =  .100000
1G000[1] =  .000000
1G010[32] =  1.00000
1G010[31] =  .978100
1G010[30] =  .931200
1G010[29] =  .514600
1G010[28] =  .858400
1G010[27] =  .399500
1G010[26] =  .778400
1G010[25] =  .332900
1G010[24] =  .692400
1G010[23] =  .284300
1G010[22] =  .646800
1G010[21] =  .259900
1G010[20] =  .599600
1G010[19] =  .241600
1G010[18] =  .503200
1G010[17] =  .206200
1G010[16] =  .446400
1G010[15] =  .190300
1G010[14] =  .390400
1G010[13] =  .175400
1G010[12] =  .332400
1G010[11] =  .161700
1G010[10] =  .272400
1G010[9] =  .148800
1G010[8] =  .208800
1G010[7] =  .136300
1G010[6] =  .142800
1G010[5] =  .123600
1G010[4] =  .073200
1G010[3] =  .111700
1G010[2] =  .036400
```

```
1G010[1] =  .106100
1G007[32] =  1.00000
1G007[31] =  .978109
1G007[30] =  .999308
1G007[29] =  .514600
1G007[28] =  .997232
1G007[27] =  .399500
1G007[26] =  .993426
1G007[25] =  .332900
1G007[24] =  .986851
1G007[23] =  .284300
1G007[22] =  .982353
1G007[21] =  .259900
1G007[20] =  .975087
1G007[19] =  .241600
1G007[18] =  .951557
1G007[17] =  .206200
1G007[16] =  .934256
1G007[15] =  .190300
1G007[14] =  .910034
1G007[13] =  .175400
1G007[12] =  .878893
1G007[11] =  .161700
1G007[10] =  .830450
1G007[9] =  .148800
1G007[8] =  .768166
1G007[7] =  .136300
1G007[6] =  .678201
1G007[5] =  .123600
1G007[4] =  .532872
1G007[3] =  .111700
1G007[2] =  .391003
1G007[1] =  .106100
1G006[32] =  1.00000
1G006[31] =  1.00000
1G006[30] =  1.00000
1G006[29] =  .666667
1G006[28] =  1.00000
1G006[27] =  .333333
1G006[26] =  1.00000
1G006[25] =  .100000
1G006[24] =  .993939
1G006[23] =  .083333
1G006[22] =  .976363
1G006[21] =  .066666
1G006[20] =  .944444
1G006[19] =  .050000
1G006[18] =  .933333
1G006[17] =  .046666
1G006[16] =  .909091
1G006[15] =  .040000
1G006[14] =  .875757
1G006[13] =  .033333
1G006[12] =  .828283
1G006[11] =  .026666
1G006[10] =  .797980
1G006[9] =  .023333
1G006[8] =  .714141
1G006[7] =  .016666
1G006[6] =  .656566
1G006[5] =  .013333
```

```
1G006[4] =  .575758
1G006[3] =  .010000
1G006[2] =  .464646
1G006[1] =  .006666
1G005[32] = -1.00000
1G005[31] =  .215714
1G005[30] = -.757143
1G005[29] =  .214286
1G005[28] = -.485714
1G005[27] =  .210714
1G005[26] = -.445714
1G005[25] =  .207143
1G005[24] = -.404286
1G005[23] =  .200000
1G005[22] = -.372857
1G005[21] =  .192857
1G005[20] = -.347143
1G005[19] =  .185714
1G005[18] = -.328571
1G005[17] =  .178571
1G005[16] = -.311429
1G005[15] =  .171429
1G005[14] = -.281429
1G005[13] =  .157143
1G005[12] = -.260000
1G005[11] =  .142857
1G005[10] = -.202857
1G005[9] =  .107143
1G005[8] = -.142857
1G005[7] =  .071428
1G005[6] = -.785714
1G005[5] =  .035714
1G005[4] = -.185714
1G005[3] =  .021428
1G005[2] =  .000000
1G005[1] =  .000000
1G004[32] = -1.00000
1G004[31] =  .214286
1G004[30] = -.312500
1G004[29] =  .210714
1G004[28] = -.225000
1G004[27] =  .207143
1G004[26] = -.250000
1G004[25] =  .200000
1G004[24] = -.145000
1G004[23] =  .192857
1G004[22] = -.085000
1G004[21] =  .185714
1G004[20] = -.047500
1G004[19] =  .178571
1G004[18] = -.020000
1G004[17] =  .171429
1G004[16] =  .000000
1G004[15] =  .164286
1G004[14] =  .012500
1G004[13] =  .157143
1G004[12] =  .020000
1G004[11] =  .150000
1G004[10] =  .022500
1G004[9] =  .142857
1G004[8] =  .028750
```

```
1G004[7] =  .107143
1G004[6] =  .028750
1G004[5] =  .071428
1G004[4] =  .017500
1G004[3] =  .035714
1G004[2] =  .000000
1G004[1] =  .000000
1G003[32] =  1.00000
1G003[31] =  .950000
1G003[30] =  .943150
1G003[29] =  .860000
1G003[28] =  .897484
1G003[27] =  .830000
1G003[26] =  .843430
1G003[25] =  .800000
1G003[24] =  .738117
1G003[23] =  .750000
1G003[22] =  .608574
1G003[21] =  .700000
1G003[20] =  .477167
1G003[19] =  .650000
1G003[18] =  .356477
1G003[17] =  .600000
1G003[16] =  .256291
1G003[15] =  .550000
1G003[14] =  .172414
1G003[13] =  .500000
1G003[12] =  .097856
1G003[11] =  .450000
1G003[10] =  .032618
1G003[9] =  .400000
1G003[8] = -.009319
1G003[7] =  .350000
1G003[6] = -.046598
1G003[5] =  .300000
1G003[4] = -.066635
1G003[3] =  .150000
1G003[2] =  .000000
1G003[1] =  .000000
1G002[32] =  1.00000
1G002[31] =  .950000
1G002[30] =  .900471
1G002[29] =  .860000
1G002[28] =  .819099
1G002[27] =  .830000
1G002[26] =  .741762
1G002[25] =  .800000
1G002[24] =  .692670
1G002[23] =  .750000
1G002[22] =  .623403
1G002[21] =  .700000
1G002[20] =  .527236
1G002[19] =  .650000
1G002[18] =  .470074
1G002[17] =  .600000
1G002[16] =  .418964
1G002[15] =  .550000
1G002[14] =  .358440
1G002[13] =  .500000
1G002[12] =  .342972
1G002[11] =  .450000
```

```
1G002[10] =  .315400
1G002[9] =  .400000
1G002[8] =  .272360
1G002[7] =  .350000
1G002[6] =  .244116
1G002[5] =  .300000
1G002[4] =  .137189
1G002[3] =  .150000
1G002[2] =  .000000
1G002[1] =  .000000
1G001[32] =  1.00000
1G001[31] =  .950000
1G001[30] =  .944700
1G001[29] =  .860000
1G001[28] =  .898618
1G001[27] =  .830000
1G001[26] =  .838710
1G001[25] =  .800000
1G001[24] =  .738249
1G001[23] =  .750000
1G001[22] =  .610138
1G001[21] =  .700000
1G001[20] =  .483871
1G001[19] =  .650000
1G001[18] =  .368664
1G001[17] =  .600000
1G001[16] =  .267281
1G001[15] =  .550000
1G001[14] =  .188018
1G001[13] =  .500000
1G001[12] =  .117051
1G001[11] =  .450000
1G001[10] =  .061751
1G001[9] =  .400000
1G001[8] =  .022119
1G001[7] =  .350000
1G001[6] = -.012903
1G001[5] =  .300000
1G001[4] = -.041474
1G001[3] =  .150000
1G001[2] =  .000000
1G001[1] =  .000000
U001 =  .050000
L001 =  .050000
1C000 =  .200000
1C063 =  .809315
1C062 =  1.00000
1C061 =  .350000
1C060 =  1.00000
1C058 =  1.00000
1C057 =  .381150
1C056 =  .381150
1C055 =  .226500
1C053 =  .226500
1C052 =  .098133
1C051 =  .333333
1C050 =  .700000
1C048 =  1.00000
1C047 =  .999999
1C046 =  .999999
1C045 =  .791459
```

```
1C043 =  1.00000
1C042 =  .142857
1C041 =  .100267
1C040 =  .468300
1C038 =  1.00000
1C037 =  1.00000
1C036 =  1.00000
1C035 =  .100000
1C033 =  1.00000
1C032 =  .710473
1C031 =  .105027
1C030 =  .099999
1C028 =  1.00000
1C027 =  .333333
1C026 =  1.00000
1C025 =  .301630
1C023 =  .152400
1C022 =  .120000
1C021 =  .200000
1C020 =  .166667
1C018 =  .080000
1C017 =  .095287
1C016 =  .403253
1C015 =  .346000
1C013 =  .242200
1C012 =  .500000
1C011 =  .514502
1C010 =  .113250
1C008 =  .142857
1C007 =  .398933
1C006 =  .766429
1C005 =  .950000
1C004 =  .100000
1C003 =  .997333
1C002 =  .020000
1C001 =  .775000
↑AAV018 =  .899997*10** 07
↑AAV017 =  .899997*10** 07
↑AAV012 =  54.7722
↑AAV007 =  .224999*10** 07
P4N =  4.28146
DER005 = -5000.00
DER004 =  999.999
NECS =  500.000
WF =  1.23440
DWF =  .320000
DER003 =  470.463
AAV020 =  470.463
AAV019 =  697.573
DH43 =  7.44793
NC =  7999.99
DH4 =  87.6352
AAV018 =  .212226*10** 07
AAV017 =  .318333*10** 07
T5 =  1456.80
DENT =  1.22474
AAV016 =  .224734
AAV015 =  1.02556
AAV014 =  .219135
DER002 = -6211.51
WCLP =  87.4484
```

```
AAV013 =  3693.80
AAV012 =  42.2397
AAV011 =  28.8350
T4 =  1784.19
AAV010 =  936.370
AAV009 =  1.22206
AAV008 =  .990000
P4 =  128.102
DH3 =  80.1872
AAV007 =  718793.
WCF6 =  28.3600
AAV006 =  65.3000
T3 =  847.818
AAV005 = -.293440
AAV004 =  .927000
WCF3 =  27.4600
AAV003 =  66.2000
P3 =  134.844
AAV002 = -.773200
AAV001 =  5.28000
DER001 =  1.00000
F9[16] =  .250000
F9[15] =  .232800
F9[14] =  .214600
F9[13] =  .194600
F9[12] =  .173100
F9[11] =  .161700
F9[10] =  .149900
F9[9] =  .125800
F9[8] =  .111600
F9[7] =  .097600
F9[6] =  .083100
F9[5] =  .068100
F9[4] =  .052200
F9[3] =  .035700
F9[2] =  .018300
F9[1] =  .009100
DMP4P2[16] =  9.78100
DMP4P2[15] =  5.14600
DMP4P2[14] =  3.99500
DMP4P2[13] =  3.32900
DMP4P2[12] =  2.84300
DMP4P2[11] =  2.59900
DMP4P2[10] =  2.41600
DMP4P2[9] =  2.06200
DMP4P2[8] =  1.90300
DMP4P2[7] =  1.75400
DMP4P2[6] =  1.61700
DMP4P2[5] =  1.48800
DMP4P2[4] =  1.36300
DMP4P2[3] =  1.23600
DMP4P2[2] =  1.11700
DMP4P2[1] =  1.06100
F8[16] =  28.9000
F8[15] =  28.8800
F8[14] =  28.8200
F8[13] =  28.7100
F8[12] =  28.5200
F8[11] =  28.3900
F8[10] =  28.1800
F8[9] =  27.5000
```

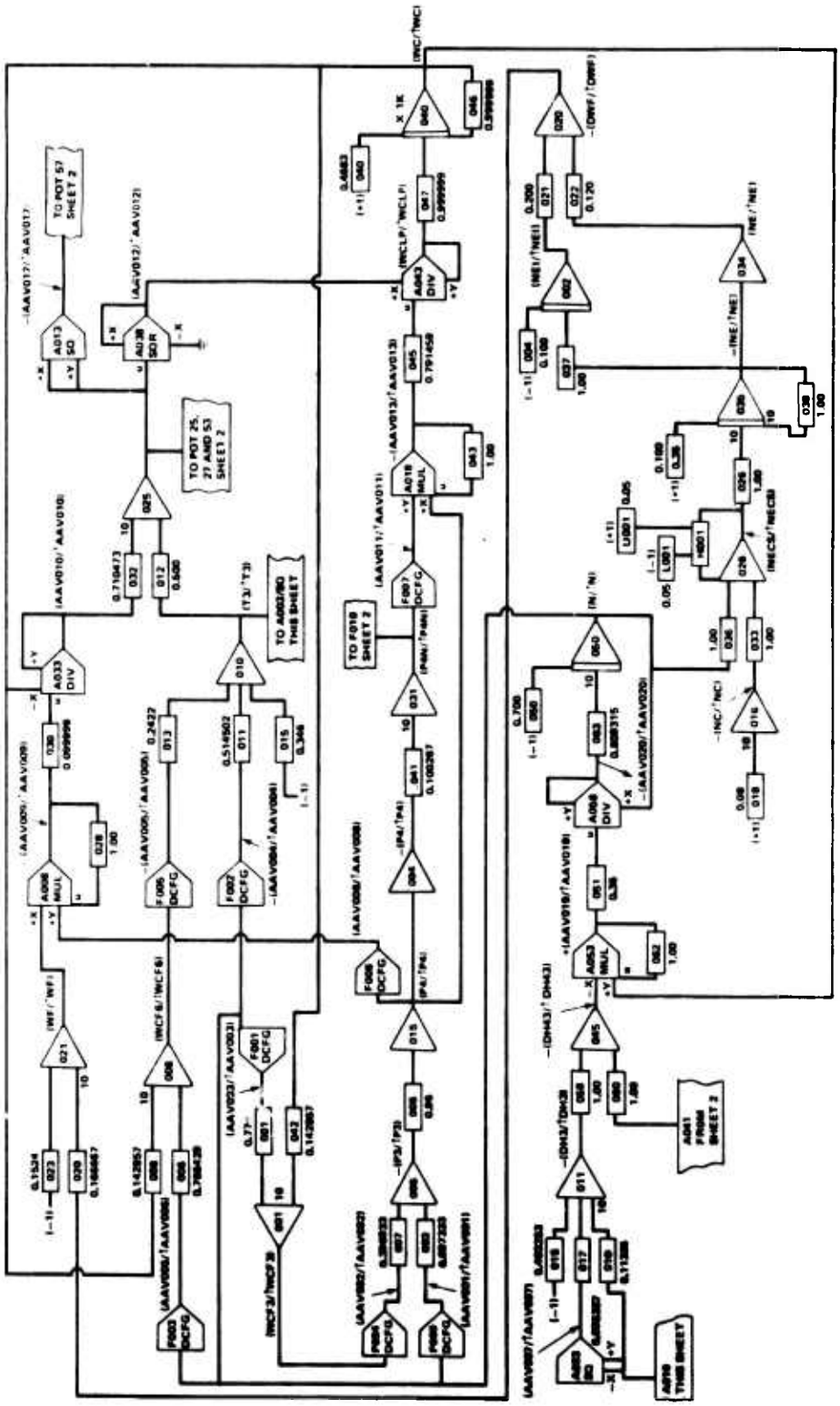
```
F8[8] =  27.0000
F8[7] =  26.3000
F8[6] =  25.4000
F8[5] =  24.0000
F8[4] =  22.2000
F8[3] =  19.6000
F8[2] =  15.4000
F8[1] =  11.3000
DMP4[16] =  300.000
DMP4[15] =  200.000
DMP4[14] =  100.000
DMP4[13] =  30.0000
DMP4[12] =  25.0000
DMP4[11] =  20.0000
DMP4[10] =  15.0000
DMP4[9] =  14.0000
DMP4[8] =  12.0000
DMP4[7] =  10.0000
DMP4[6] =  8.00000
DMP4[5] =  7.00000
DMP4[4] =  5.00000
DMP4[3] =  4.00000
DMP4[2] =  3.00000
DMP4[1] =  2.00000
F7[16] =  .990000
F7[15] =  .990000
F7[14] =  .990000
F7[13] =  .990000
F7[12] =  .984000
F7[11] =  .966600
F7[10] =  .935000
F7[9] =  .924000
F7[8] =  .910000
F7[7] =  .867000
F7[6] =  .820000
F7[5] =  .790000
F7[4] =  .707000
F7[3] =  .650000
F7[2] =  .570000
F7[1] =  .460000
DMWCF6[16] =  30.2000
DMWCF6[15] =  30.0000
DMWCF6[14] =  29.5000
DMWCF6[13] =  29.0000
DMWCF6[12] =  28.0000
DMWCF6[11] =  27.0000
DMWCF6[10] =  26.0000
DMWCF6[9] =  25.0000
DMWCF6[8] =  24.0000
DMWCF6[7] =  22.0000
DMWCF6[6] =  20.0000
DMWCF6[5] =  15.0000
DMWCF6[4] =  10.0000
DMWCF6[3] =  5.00000
DMWCF6[2] =  3.00000
DMWCF6[1] =  .000000
F5[16] = -.700000
F5[15] = -.530000
F5[14] = -.340000
F5[13] = -.312000
F5[12] = -.283000
```

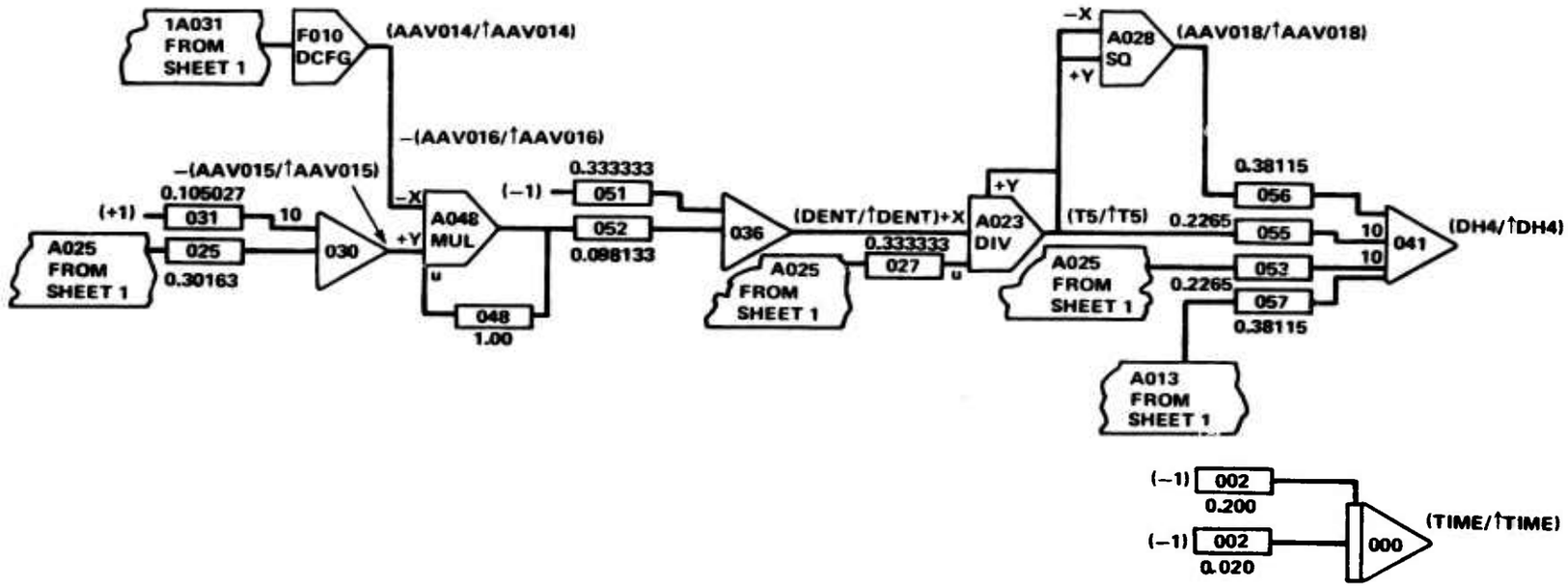
```
F5[11] = -.261000
F5[10] = -.243000
F5[9] = -.230000
F5[8] = -.218000
F5[7] = -.197000
F5[6] = -.182000
F5[5] = -.142000
F5[4] = -.100000
F5[3] = -.550000
F5[2] = -.340000
F5[1] =  .000000
DMWCF3[16] =  30.0000
DMWCF3[15] =  29.5000
DMWCF3[14] =  29.0000
DMWCF3[13] =  28.0000
DMWCF3[12] =  27.0000
DMWCF3[11] =  26.0000
DMWCF3[10] =  25.0000
DMWCF3[9] =  24.0000
DMWCF3[8] =  23.0000
DMWCF3[7] =  22.0000
DMWCF3[6] =  21.0000
DMWCF3[5] =  20.0000
DMWCF3[4] =  15.0000
DMWCF3[3] =  10.0000
DMWCF3[2] =  5.00000
DMWCF3[1] =  .000000
F2[16] = -4.00000
F2[15] = -3.25000
F2[14] = -2.50000
F2[13] = -1.00000
F2[12] = -.580000
F2[11] = -.340000
F2[10] = -.190000
F2[9] = -.080000
F2[8] =  .000000
F2[7] =  .050000
F2[6] =  .080000
F2[5] =  .090000
F2[4] =  .115000
F2[3] =  .115000
F2[2] =  .070000
F2[1] =  .000000
F6[16] =  107.300
F6[15] =  101.200
F6[14] =  96.3000
F6[13] =  90.5000
F6[12] =  79.2000
F6[11] =  65.3000
F6[10] =  51.2000
F6[9] =  38.2500
F6[8] =  27.5000
F6[7] =  18.5000
F6[6] =  10.5000
F6[5] =  3.50000
F6[4] = -1.00000
F6[3] = -5.00000
F6[2] = -7.15000
F6[1] =  .000000
F4[16] =  1.48700
F4[15] =  1.33900
```

```
F4[14] =  1.21800
F4[13] =  1.10300
F4[12] =  1.03000
F4[11] =  .927000
F4[10] =  .784000
F4[9] =  .699000
F4[8] =  .623000
F4[7] =  .533000
F4[6] =  .510000
F4[5] =  .469000
F4[4] =  .405000
F4[3] =  .363000
F4[2] =  .204000
F4[1] =  .000000
F3[16] =  108.500
F3[15] =  102.500
F3[14] =  97.5000
F3[13] =  90.9999
F3[12] =  80.0999
F3[11] =  66.2000
F3[10] =  52.5000
F3[9] =  40.0000
F3[8] =  29.0000
F3[7] =  20.4000
F3[6] =  12.7000
F3[5] =  6.70000
F3[4] =  2.40000
F3[3] = -1.40000
F3[2] = -4.50000
F3[1] =  .000000
DUMN[16] =  9499.99
DUMN[15] =  8599.99
DUMN[14] =  8299.99
DUMN[13] =  7999.99
DUMN[12] =  7499.99
DUMN[11] =  6999.99
DUMN[10] =  6499.99
DUMN[9] =  6000.00
DUMN[8] =  5500.00
DUMN[7] =  5000.00
DUMN[6] =  4500.00
DUMN[5] =  4000.00
DUMN[4] =  3500.00
DUMN[3] =  3000.00
DUMN[2] =  1500.00
DUMN[1] =  .000000
F1[16] =  10.0000
F1[15] =  9.16000
F1[14] =  8.23000
F1[13] =  7.33000
F1[12] =  6.46000
F1[11] =  5.28000
F1[10] =  3.76000
F1[9] =  2.93000
F1[8] =  2.39000
F1[7] =  2.03000
F1[6] =  1.74000
F1[5] =  1.54000
F1[4] =  1.34000
F1[3] =  1.20000
F1[2] =  .970000
```

```
F1[1] =  1.00000
NE =  999.999
NEI =  999.999
Y =  1.00000
CON =  1.00000
GO =  .000000
↑NECS =  9999.99
↑WF =  6.00000
↑DWF =  10.0000
↑AAV020 =  80931.4
↑AAV019 =  60000.0
↑DH43 =  300.000
↑NC =  9999.99
↑DH4 =  300.000
↑T5 =  3000.00
↑DENT =  3.00000
↑AAV016 =  .294400
↑AAV015 =  1.17760
↑AAV014 =  .250000
↑WCLP =  200.000
↑AAV013 =  8669.99
↑AAV011 =  28.9000
↑T4 =  3000.00
↑AAV010 =  21314.2
↑AAV009 =  5.94000
↑AAV008 =  .990000
↑P4 =  300.000
↑DH3 =  300.000
↑WCF6 =  140.000
↑AAV006 =  107.300
↑T3 =  1500.00
↑AAV005 =  .700000
↑AAV004 =  1.48700
↑WCF3 =  140.000
↑AAV003 =  108.500
↑P3 =  300.000
↑AAV002 =  4.00000
↑AAV001 =  10.0000
↑NE =  9999.99
↑NEI =  9999.99
↑N =  9999.99
↑WC =  200.000
↑TIME =  50.0000
K2 =  2.00000
K1 =  1.20000
NCOM =  .000000
WFO =  .914400
NCO =  7999.99
1P000 =  -.200000
1P063 =  -.004704
1F007 =  .997751
N =  .000000
TIME =  10.0000
↑P4N =  10.0000
WC =  93.6600
BETA =  1.00000
```

2.0 ANALOG DIAGRAMS




1A031
FROM
SHEET 1
F010
DCFG
(AAV014/↑AAV014)
–(AAV016/↑AAV016)
–(AAV015/↑AAV015)
0.105027
(+1)
031
10
A025
FROM
SHEET 1
025
0.30163
030
–X
+Y
A048
MUL
u
048
1.00
0.333333
(–1)
051
052
0.098133
036
A025
FROM
SHEET 1
(DENT/↑DENT)+X
0.333333
027
u
+Y
A023
DIV
–X
A028
SQ
+Y
(AAV018/↑AAV018)
(T5/↑T5)
0.38115
056
0.2265
055
10
053
10
041
(DH4/↑DH4)
A025
FROM
SHEET 1
0.2265
057
0.38115
A013
FROM
SHEET 1
(–1)
002
0.200
(–1)
002
0.020
000
(TIME/↑TIME)

3.0 FUNCTION LISTINGS

TYPICAL FUNCTION DIRECTORY

MVFG DIRECTORY FILE = J65AD:

FUNCTION NAME	LOG. UNIT	MVFG MODULE	MVFG MODE	OUTPUT HOLE	FCT. NO.	FUNCTION--VARIABLE VARIABLE	--NAME	VARIABLE SCALE FACTOR
P3R	1	1	2	1	1	OUTPUT	AAV001	.925199890E+01
						1	WC	.199999969E+03
						2	N	.999999609E+04
						3		.000000000E+00
F4	0	0	1	1	1	OUTPUT	AAV002	.148699999E+01
						1	N	.999999609E+04
						2		.000000000E+00
						3		.000000000E+00
F5	0	0	1	2	2	OUTPUT	AAV003	.700000048E+00
						1	WCF6	.139999939E+03
						2		.000000000E+00
						3		.000000000E+00
F6	0	0	1	5	5	OUTPUT	AAV004	.107299973E+03
						1	N	.999999609E+04
						2		.000000000E+00
						3		.000000000E+00
F7	0	0	1	3	3	OUTPUT	AAV006	.990000010E+00
						1	P4	.299999939E+03
						2		.000000000E+00
						3		.000000000E+00
F8	0	0	1	4	4	OUTPUT	AAV009	.288999901E+02
						1	P4N	.999999609E+01
						2		.000000000E+00
						3		.000000000E+00
F9	0	0	1	8	8	OUTPUT	AAV012	.250000000E+00
						1	P4N	.999999609E+01
						2		.000000000E+00
						3		.000000000E+00

USER NAME - TYPICAL FUNCTION DATA FOR D3R (WC,N)

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	P3R		.100000000E+02	
VARIABLE				
1	WC		.200000000E+03	55
2	N		.100000000E+05	5

	N	WC	P3R	INDEX
	.50000E+04	.30000E+02	.48530E+01	1, 1,
		.35000E+02	.48530E+01	2, 1,
		.40000E+02	.48530E+01	3, 1,
		.45000E+02	.48530E+01	4, 1,
		.50000E+02	.48530E+01	5, 1,
		.55000E+02	.48530E+01	6, 1,
		.60000E+02	.48530E+01	7, 1,
		.65000E+02	.48530E+01	8, 1,
		.70000E+02	.48530E+01	9, 1,
		.75000E+02	.48530E+01	10, 1,
		.80000E+02	.48530E+01	11, 1,
		.81000E+02	.48530E+01	12, 1,
		.82000E+02	.48500E+01	13, 1,
		.83000E+02	.48460E+01	14, 1,
		.84000E+02	.48400E+01	15, 1,
		.85000E+02	.48310E+01	16, 1,
		.86000E+02	.48220E+01	17, 1,
		.87000E+02	.47980E+01	18, 1,
		.88000E+02	.47560E+01	19, 1,
		.89000E+02	.46800E+01	20, 1,
		.90000E+02	.45940E+01	21, 1,
		.91000E+02	.44640E+01	22, 1,
		.92000E+02	.42610E+01	23, 1,
		.93000E+02	.39080E+01	24, 1,
		.94000E+02	.27180E+01	25, 1,
		.95000E+02	.13610E+01	26, 1,
		.96000E+02	.10000E+01	27, 1,
		.97000E+02	.10000E+01	28, 1,
		.98000E+02	.10000E+01	29, 1,
		.99000E+02	.10000E+01	30, 1,
		.10000E+03	.10000E+01	31, 1,
		.10100E+03	.10000E+01	32, 1,
		.10200E+03	.10000E+01	33, 1,
		.10300E+03	.10000E+01	34, 1,
		.10400E+03	.10000E+01	35, 1,
		.10500E+03	.10000E+01	36, 1,
		.10600E+03	.10000E+01	37, 1,
		.10700E+03	.10000E+01	38, 1,
		.10800E+03	.10000E+01	39, 1,
		.10900E+03	.10000E+01	40, 1,
		.11000E+03	.10000E+01	41, 1,
		.11100E+03	.10000E+01	42, 1,
		.11200E+03	.10000E+01	43, 1,
		.11300E+03	.10000E+01	44, 1,
		.11400E+03	.10000E+01	45, 1,
		.11500E+03	.10000E+01	46, 1,

```
                    .11600E+03      .10000E+01  47, 1,
                    .11700E+03      .10000E+01  48, 1,
                    .11800E+03      .10000E+01  49, 1,
                    .11900E+03      .10000E+01  50, 1,
                    .12000E+03      .10000E+01  51, 1,
                    .12100E+03      .10000E+01  52, 1,
                    .12200E+03      .10000E+01  53, 1,
                    .12300E+03      .10000E+01  54, 1,
                    .12400E+03      .10000E+01  55, 1,
.70000E+04          .30000E+02      .53920E+01   1, 2,
                    .35000E+02      .53920E+01   2, 2,
                    .40000E+02      .53920E+01   3, 2,
                    .45000E+02      .53920E+01   4, 2,
                    .50000E+02      .53920E+01   5, 2,
                    .55000E+02      .53920E+01   6, 2,
                    .60000E+02      .53920E+01   7, 2,
                    .65000E+02      .53920E+01   8, 2,
                    .70000E+02      .53920E+01   9, 2,
                    .75000E+02      .53920E+01  10, 2,
                    .80000E+02      .53920E+01  11, 2,
                    .81000E+02      .53920E+01  12, 2,
                    .82000E+02      .53880E+01  13, 2,
                    .83000E+02      .53840E+01  14, 2,
                    .84000E+02      .53780E+01  15, 2,
                    .85000E+02      .53680E+01  16, 2,
                    .86000E+02      .53580E+01  17, 2,
                    .87000E+02      .53320E+01  18, 2,
                    .88000E+02      .52840E+01  19, 2,
                    .89000E+02      .52000E+01  20, 2,
                    .90000E+02      .51040E+01  21, 2,
                    .91000E+02      .49600E+01  22, 2,
                    .92000E+02      .47340E+01  23, 2,
                    .93000E+02      .43420E+01  24, 2,
                    .94000E+02      .30200E+01  25, 2,
                    .95000E+02      .15120E+01  26, 2,
                    .96000E+02      .10000E+01  27, 2,
                    .97000E+02      .10000E+01  28, 2,
                    .98000E+02      .10000E+01  29, 2,
                    .99000E+02      .10000E+01  30, 2,
                    .10000E+03      .10000E+01  31, 2,
                    .10100E+03      .10000E+01  32, 2,
                    .10200E+03      .10000E+01  33, 2,
                    .10300E+03      .10000E+01  34, 2,
                    .10400E+03      .10000E+01  35, 2,
                    .10500E+03      .10000E+01  36, 2,
                    .10600E+03      .10000E+01  37, 2,
                    .10700E+03      .10000E+01  38, 2,
                    .10800E+03      .10000E+01  39, 2,
                    .10900E+03      .10000E+01  40, 2,
                    .11000E+03      .10000E+01  41, 2,
                    .11100E+03      .10000E+01  42, 2,
                    .11200E+03      .10000E+01  43, 2,
                    .11300E+03      .10000E+01  44, 2,
                    .11400E+03      .10000E+01  45, 2,
                    .11500E+03      .10000E+01  46, 2,
                    .11600E+03      .10000E+01  47, 2,
                    .11700E+03      .10000E+01  48, 2,
                    .11800E+03      .10000E+01  49, 2,
                    .11900E+03      .10000E+01  50, 2,
                    .12000E+03      .10000E+01  51, 2,
                    .12100E+03      .10000E+01  52, 2,
```

```
                .12200E+03      .10000E+01  53, 2,
                .12300E+03      .10000E+01  54, 2,
                .12400E+03      .10000E+01  55, 2,
.75000E+04      .30000E+02      .65720E+01   1, 3,
                .35000E+02      .65720E+01   2, 3,
                .40000E+02      .65720E+01   3, 3,
                .45000E+02      .65720E+01   4, 3,
                .50000E+02      .65720E+01   5, 3,
                .55000E+02      .65720E+01   6, 3,
                .60000E+02      .65720E+01   7, 3,
                .65000E+02      .65720E+01   8, 3,
                .70000E+02      .65720E+01   9, 3,
                .75000E+02      .65720E+01  10, 3,
                .80000E+02      .65720E+01  11, 3,
                .81000E+02      .65720E+01  12, 3,
                .82000E+02      .65720E+01  13, 3,
                .83000E+02      .65720E+01  14, 3,
                .84000E+02      .65720E+01  15, 3,
                .85000E+02      .65720E+01  16, 3,
                .86000E+02      .65720E+01  17, 3,
                .87000E+02      .65720E+01  18, 3,
                .88000E+02      .65720E+01  19, 3,
                .89000E+02      .65720E+01  20, 3,
                .90000E+02      .65720E+01  21, 3,
                .91000E+02      .65720E+01  22, 3,
                .92000E+02      .65720E+01  23, 3,
                .93000E+02      .65720E+01  24, 3,
                .94000E+02      .65720E+01  25, 3,
                .95000E+02      .65720E+01  26, 3,
                .96000E+02      .65680E+01  27, 3,
                .97000E+02      .65640E+01  28, 3,
                .98000E+02      .65580E+01  29, 3,
                .99000E+02      .65540E+01  30, 3,
                .10000E+03      .65480E+01  31, 3,
                .10100E+03      .65380E+01  32, 3,
                .10200E+03      .65100E+01  33, 3,
                .10300E+03      .64620E+01  34, 3,
                .10400E+03      .63860E+01  35, 3,
                .10500E+03      .62760E+01  36, 3,
                .10600E+03      .61300E+01  37, 3,
                .10700E+03      .58980E+01  38, 3,
                .10800E+03      .54920E+01  39, 3,
                .10900E+03      .41000E+01  40, 3,
                .11000E+03      .26000E+01  41, 3,
                .11100E+03      .26000E+01  42, 3,
                .11200E+03      .26000E+01  43, 3,
                .11300E+03      .26000E+01  44, 3,
                .11400E+03      .26000E+01  45, 3,
                .11500E+03      .26000E+01  46, 3,
                .11600E+03      .26000E+01  47, 3,
                .11700E+03      .26000E+01  48, 3,
                .11800E+03      .26000E+01  49, 3,
                .11900E+03      .26000E+01  50, 3,
                .12000E+03      .26000E+01  51, 3,
                .12100E+03      .26000E+01  52, 3,
                .12200E+03      .26000E+01  53, 3,
                .12300E+03      .26000E+01  54, 3,
                .12400E+03      .26000E+01  55, 3,
.80000E+04      .30000E+02      .74420E+01   1, 4,
                .35000E+02      .74420E+01   2, 4,
                .40000E+02      .74420E+01   3, 4,
```

```
                .45000E+02      .74420E+01    4, 4,
                .50000E+02      .74420E+01    5, 4,
                .55000E+02      .74420E+01    6, 4,
                .60000E+02      .74420E+01    7, 4,
                .65000E+02      .74420E+01    8, 4,
                .70000E+02      .74420E+01    9, 4,
                .75000E+02      .74420E+01   10, 4,
                .80000E+02      .74420E+01   11, 4,
                .81000E+02      .74420E+01   12, 4,
                .82000E+02      .74420E+01   13, 4,
                .83000E+02      .74420E+01   14, 4,
                .84000E+02      .74420E+01   15, 4,
                .85000E+02      .74420E+01   16, 4,
                .86000E+02      .74420E+01   17, 4,
                .87000E+02      .74420E+01   18, 4,
                .88000E+02      .74420E+01   19, 4,
                .89000E+02      .74420E+01   20, 4,
                .90000E+02      .74420E+01   21, 4,
                .91000E+02      .74420E+01   22, 4,
                .92000E+02      .74420E+01   23, 4,
                .93000E+02      .74420E+01   24, 4,
                .94000E+02      .74420E+01   25, 4,
                .95000E+02      .74420E+01   26, 4,
                .96000E+02      .74420E+01   27, 4,
                .97000E+02      .74420E+01   28, 4,
                .98000E+02      .74420E+01   29, 4,
                .99000E+02      .74420E+01   30, 4,
                .10000E+03      .74420E+01   31, 4,
                .10100E+03      .74420E+01   32, 4,
                .10200E+03      .74420E+01   33, 4,
                .10300E+03      .74420E+01   34, 4,
                .10400E+03      .74420E+01   35, 4,
                .10500E+03      .74420E+01   36, 4,
                .10600E+03      .74420E+01   37, 4,
                .10700E+03      .74380E+01   38, 4,
                .10800E+03      .74340E+01   39, 4,
                .10900E+03      .74280E+01   40, 4,
                .11000E+03      .74240E+01   41, 4,
                .11100E+03      .74200E+01   42, 4,
                .11200E+03      .74080E+01   43, 4,
                .11300E+03      .73780E+01   44, 4,
                .11400E+03      .73260E+01   45, 4,
                .11500E+03      .72460E+01   46, 4,
                .11600E+03      .71320E+01   47, 4,
                .11700E+03      .69800E+01   48, 4,
                .11800E+03      .67420E+01   49, 4,
                .11900E+03      .62880E+01   50, 4,
                .12000E+03      .48000E+01   51, 4,
                .12100E+03      .48000E+01   52, 4,
                .12200E+03      .48000E+01   53, 4,
                .12300E+03      .48000E+01   54, 4,
                .12400E+03      .48000E+01   55, 4,
.86000E+04      .30000E+02      .81860E+01    1, 5,
                .35000E+02      .81860E+01    2, 5,
                .40000E+02      .81860E+01    3, 5,
                .45000E+02      .81860E+01    4, 5,
                .50000E+02      .81860E+01    5, 5,
                .55000E+02      .81860E+01    6, 5,
                .60000E+02      .81860E+01    7, 5,
                .65000E+02      .81860E+01    8, 5,
                .70000E+02      .81860E+01    9, 5,
```

```
.75000E+02     .81860E+01  10, 5,
.80000E+02     .81860E+01  11, 5,
.81000E+02     .81860E+01  12, 5,
.82000E+02     .81860E+01  13, 5,
.83000E+02     .81860E+01  14, 5,
.84000E+02     .81860E+01  15, 5,
.85000E+02     .81860E+01  16, 5,
.86000E+02     .81860E+01  17, 5,
.87000E+02     .81860E+01  18, 5,
.88000E+02     .81860E+01  19, 5,
.89000E+02     .81860E+01  20, 5,
.90000E+02     .81860E+01  21, 5,
.91000E+02     .81860E+01  22, 5,
.92000E+02     .81860E+01  23, 5,
.93000E+02     .81860E+01  24, 5,
.94000E+02     .81860E+01  25, 5,
.95000E+02     .81860E+01  26, 5,
.96000E+02     .81860E+01  27, 5,
.97000E+02     .81860E+01  28, 5,
.98000E+02     .81860E+01  29, 5,
.99000E+02     .81860E+01  30, 5,
.10000E+03     .81860E+01  31, 5,
.10100E+03     .81860E+01  32, 5,
.10200E+03     .81860E+01  33, 5,
.10300E+03     .81860E+01  34, 5,
.10400E+03     .81860E+01  35, 5,
.10500E+03     .81860E+01  36, 5,
.10600E+03     .81860E+01  37, 5,
.10700E+03     .81820E+01  38, 5,
.10800E+03     .81770E+01  39, 5,
.10900E+03     .81710E+01  40, 5,
.11000E+03     .81660E+01  41, 5,
.11100E+03     .81620E+01  42, 5,
.11200E+03     .81490E+01  43, 5,
.11300E+03     .81160E+01  44, 5,
.11400E+03     .80590E+01  45, 5,
.11500E+03     .79710E+01  46, 5,
.11600E+03     .78450E+01  47, 5,
.11700E+03     .76780E+01  48, 5,
.11800E+03     .71460E+01  49, 5,
.11900E+03     .69170E+01  50, 5,
.12000E+03     .52800E+01  51, 5,
.12100E+03     .52800E+01  52, 5,
.12200E+03     .52800E+01  53, 5,
.12300E+03     .52800E+01  54, 5,
.12400E+03     .52800E+01  55, 5,

END OF FUNCTION TABLE
```

TYPICAL FUNCTION DATA FOR F4(N)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	F4		.200000000E+01	
VARIABLE 1	N		.100000000E+05	16

		N	F4	INDEX
		.00000E+00	.00000E+00	1,
		.15000E+04	.20400E+00	2,
		.30000E+04	.36300E+00	3,
		.35000E+04	.40500E+00	4,
		.40000E+04	.46900E+00	5,
		.45000E+04	.51000E+00	6,
		.50000E+04	.53300E+00	7,
		.55000E+04	.62300E+00	8,
		.60000E+04	.69900E+00	9,
		.65000E+04	.78400E+00	10,
		.70000E+04	.92700E+00	11,
		.75000E+04	.10300E+01	12,
		.80000E+04	.11030E+01	13,
		.83000E+04	.12180E+01	14,
		.86000E+04	.13390E+01	15,
		.95000E+04	.14870E+01	16,

END OF FUNCTION TABLE

TYPICAL FUNCTION DATA FOR F5 (WCF6)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	F5		.100000000E+01	
VARIABLE 1	WCF6		.400000000E+02	16

		WCF6	F5	INDEX
		.00000E+00	.00000E+00	1,
		.30000E+01	-.34000E+00	2,
		.50000E+01	-.55000E+00	3,
		.10000E+02	-.10000E+00	4,
		.15000E+02	-.14200E+00	5,
		.20000E+02	-.18200E+00	6,
		.22000E+02	-.19700E+00	7,
		.24000E+02	-.21800E+00	8,
		.25000E+02	-.23000E+00	9,
		.26000E+02	-.24300E+00	10,
		.27000E+02	-.26100E+00	11,
		.28000E+02	-.28300E+00	12,
		.29000E+02	-.31200E+00	13,
		.29500E+02	-.34000E+00	14,
		.30000E+02	-.53000E+00	15,
		.30200E+02	-.70000E+00	16,

END OF FUNCTION TABLE

TYPICAL FUNCTION DATA FOR F6 (N)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	F6		.200000000E+03	
VARIABLE 1	N		.100000000E+05	16

		N	F6	INDEX
		.00000E+00	.00000E+00	1,
		.15000E+04	-.71500E+01	2,
		.30000E+04	-.50000E+01	3,
		.35000E+04	-.10000E+01	4,
		.40000E+04	.35000E+01	5,
		.45000E+04	.10500E+02	6,
		.50000E+04	.18500E+02	7,
		.55000E+04	.27500E+02	8,
		.60000E+04	.38250E+02	9,
		.65000E+04	.51200E+02	10,
		.70000E+04	.65300E+02	11,
		.75000E+04	.79200E+02	12,
		.80000E+04	.90500E+02	13,
		.83000E+04	.96300E+02	14,
		.86000E+04	.10120E+03	15,
		.95000E+04	.10730E+03	16,

END OF FUNCTION TABLE

TYPICAL FUNCTION DATA FOR F7 (P4)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. DPTS
FUNCTION	F7		.100000000E+01	
VARIABLE 1	P4		.400000000E+03	16

		P4	F7	INDEX
		.20000E+01	.46000E+00	1,
		.30000E+01	.57000E+00	2,
		.40000E+01	.65000E+00	3,
		.50000E+01	.70700E+00	4,
		.70000E+01	.79000E+00	5,
		.80000E+01	.82000E+00	6,
		.10000E+02	.86700E+00	7,
		.12000E+02	.90000E+00	8,
		.14000E+02	.92400E+00	9,
		.15000E+02	.93500E+00	10,
		.20000E+02	.96660E+00	11,
		.25000E+02	.98400E+00	12,
		.30000E+02	.99000E+00	13,
		.50000E+02	.99000E+00	14,
		.10000E+03	.99000E+00	15,
		.30000E+03	.99000E+00	16,

END OF FUNCTION TABLE

TYPICAL FUNCTION DATA FOR F8 (P4N)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	F8		.400000000E+02	
VARIABLE 1	P4N		.100000000E+02	16

		P4N	F8	INDEX
		.10613E+01	.11300E+02	1,
		.11174E+01	.15400E+02	2,
		.12363E+01	.19600E+02	3,
		.13629E+01	.22200E+02	4,
		.14883E+01	.24000E+02	5,
		.16172E+01	.25400E+02	6,
		.17536E+01	.26300E+02	7,
		.19029E+01	.27000E+02	8,
		.20625E+01	.27500E+02	9,
		.24159E+01	.28180E+02	10,
		.25988E+01	.28390E+02	11,
		.28428E+01	.28520E+02	12,
		.33288E+01	.28710E+02	13,
		.39946E+01	.28820E+02	14,
		.51459E+01	.28880E+02	15,
		.97810E+01	.28900E+02	16,

END OF FUNCTION TABLE

TYPICAL FUNCTION DATA FOR F9 (P4N)

USER NAME -

	NAME	UNITS	MAXIMUM VALUE	NO. BPTS
FUNCTION	F9		.399999976E+00	
VARIABLE 1	P4N		.100000000E+02	16

		P4N	F9	INDEX
		.10613E+01	.91000E-02	1,
		.11174E+01	.18300E-01	2,
		.12363E+01	.35700E-01	3,
		.13629E-01	.52200E-01	4,
		.14883E+01	.68100E-01	5,
		.16172E+01	.83100E-01	6,
		.17536E+01	.97600E-01	7,
		.19029E+01	.11160E+00	8,
		.20625E+01	.12580E+00	9,
		.24159E+01	.14990E+00	10,
		.25988E+01	.16170E+00	11,
		.28428E+01	.17310E+00	12,
		.33288E+01	.19460E+00	13,
		.39946E+01	.21460E+00	14,
		.51459E+01	.23280E+00	15,
		.97810E+01	.25000E+00	16,

END OF FUNCTION TABLE

4.0 ANALOG STATIC CHECK RESULTS

TYPICAL OUTPUT FROM ON-LINE CONNECTION CHECK

```
                                                    3 - RUN OFF-LINE CHECK
                                                    4 - GO TO MONITOR
GO_ 1                                               5 - GO TO ANOTHER VIR MEM FILE
                                                    6 - DUMP ALL PARTS TO MAG TAPE
DAC TEST VOLTAGES SET                               7 - DUMP ALL PARTS TO SOURCE FILE ON 8
                                                    8 - MAKE LISTING OF ALL PARTS AND VARS
                                                    9 - SET TEST DAC'S
SET POTS? (Y OR N.  DEFAULT IS N)
SP_ Y
SET COEF, LIM &FUN                                  OPT_
L100 =  .050000, U100 =  .050000
HALT    IH  22.100
  G;
ON-LINE CHECK
COEF CHK
L100 =  .050000, U100 =  .050000
HALT    IH  22.100
  G;
CHECK CONNECTIONS

DERIV CHK
AMPL CHK
FUN GEN CHK
 = -.416200  VS  -.420526  IN  35.110
EXT TRK CHK
 =  .479500  VS   .484808  IN  37.220
CMP CHK

#####################  SPECIAL NOTES  ####################


------------------------------------------------------------


SELECT DESIRED OPTION FROM THE FOLLOWING LIST AND ENTER THE NUMBER
FOR THE OPTION - DEFAULT IS ON-LINE SETUP AND CONNECTION CHECK

     0 - ON-LINE SETUP AND CONNECTION CHECK
     1 - ON-LINE EQUATION CHECK
     2 - SETUP FOR PRODUCTION
```

TYPICAL OUTPUT FROM ON-LINE EQUATION CHECK

```
1
ON-LINE CHECK AGAINST EQUATIONS
PARAM & STATIC TEST VAL OF VARS
LU0_
LU1_
LU2_
LU3_
LU4_
LU5_
LU6_
LU7_
AUX PARAM
IS ANALOG COMPUTER ON-LINE?
GO_

CHECKING TEST DAC CONNECTIONS


TEST MDAC INPUT - CON M41 TO V02

PATCH T/S OR ENTER 1 FOR GO TO BYPASS T/S PATCH CHECK

GO_ 1

DAC TEST VOLTAGES SET

DERIVATIVES
 =  .818460  VS   .813374  IN  42.812
AMP CHK
 =  .268400  VS   .267291  IN  43.011
 =  .421260  VS   .426117  IN  43.033
 =  .442000  VS   .447350  IN  43.102
 =  .320500  VS   .319464  IN  43.103
 =  .420400  VS   .424983  IN  43.112
 =  .418700  VS   .424059  IN  43.115
 =  .450200  VS   .454925  IN  43.123
 =  .773100  VS   .775598  IN  43.124
 = -.420300  VS  -.424983  IN  43.131
 = -.001200  VS  -.000855  IN  43.134
 = -.872000  VS  -.875256  IN  43.141
 = -.002300  VS  -.000117  IN  43.302
FUN GEN CHK
EXT TRK CHK
 =  .470900  VS   .484000  IN  47.220
CMP CHK

EQUATION CHECK COMPLETE

####################  SPECIAL NOTES  ######

-----------------------------------------

SELECT DESIRED OPTION FROM THE FOLLOWING LI
FOR THE OPTION - DEFAULT IS ON-LINE SETUP 0

     0 - ON-LINE SETUP AND CONNECTION CHECK
     1 - ON-LINE EQUATION CHECK
     2 - SETUP FOR PRODUCTION
     3 - RUN OFF-LINE CHECK
     4 - GO TO MONITOR
     5 - GO TO ANOTHER VIR MEM FILE
     6 - DUMP ALL PARTS TO MAG TAPE
     7 - DUMP ALL PARTS TO SOURCE FILE ON D
     8 - MAKE LISTING OF ALL PARTS AND VARS
     9 - SET TEST DAC'S

OPT_
```

5.0 TRUNKING STATION CONNECTIONS

TYPICAL TRUNKING WHEN MVFG FUNCTIONS ARE REQUIRED

A N A L O G P A T C H V 1 A T O V 2 0

ANALOG 3 3 CONNECTOR V1A IS BIDIRECTIONAL
MVFG CONNECTOR V20 IS TRANSMITTING

	V1A FUNCTIONAL DESCRIPTION		V20 FUNCTIONAL DESCRIPTION
T310		V1A0	V200 - - NOT USED
T311		V1A1	V201 - - NOT USED
T312		V1A2	V202 - - NOT USED
T313		V1A3	V203 - - NOT USED
T314		V1A4	V204 - - NOT USED
T315		V1A5	V205 - - NOT USED
T316		V1A6	V206 - - NOT USED
T317	N/10^4	V1A7	V207 - - NOT USED
T330	WCF6/140	V1A8	V208 - - MV00 X(IN)
T331	P4/300	V1A9	V209 - - MV00 Y(IN)
T332	P4N/10	V1AA	V20A - - MV00 Z(IN)
T333	WC/200	V1AB	V20B - - MV00 W(IN)
T334	N/10^4	V1AC	V20C - - MV01 X(IN)
T335		V1AD	V20D - - MV01 Y(IN)
T336		V1AE	V20E - - MV01 Z(IN)
T337		V1AF	V20F - - MV01 W(IN)

A N A L O G P A T C H V 1 9 T O V 2 1

ANALOG 3 2 CONNECTOR V19 IS BIDIRECTIONAL
MVFG CONNECTOR V21 IS RECEIVING

	V19 FUNCTIONAL DESCRIPTION		V21 FUNCTIONAL DESCRIPTION
T250	.AAV002/1.487(F4(N))	V190	V210 - - MV00 F1(X) F1(X.Y) F1(X.Y.Z)
T251	.AAV003/7 (F5(WCE6))	V191	V211 - - MV00 F1(Y)
T252	.AAV006/.99 (F7(P4))	V192	V212 - - MV00 F1(Z) F1(Z.W) -F1(X.Y.Z)
T253	.AAV009/28.9 (F8(P4N))	V193	V213 - - MV00 F1(W)
T254	'AAV004/107.3 (F6(N))	V194	V214 - - MV00 F2(X) F2(X.Y) F2(X.Y.Z)
T255		V195	V215 - - MV00 F2(Y)
T256		V196	V216 - - MV00 F2(Z) F2(Z.W) -F2(X.Y.Z)
T257	AAV012/.25 (F9(P4N))	V197	V217 - - MV00 F2(W)
T270	.AAV001/9.252 (P3R(WC,N)	V198	V218 - - MV01 F1(X) F1(X.Y) F1(X.Y.Z)
T271		V199	V219 - - MV01 F1(Y)
T272		V19A	V21A - - MV01 F1(Z) F1(Z.W) -F1(X.Y.Z)
T273		V19B	V21B - - MV01 F1(W)
T274		V19C	V21C - - MV01 F2(X) F2(X.Y) F2(X.Y.Z)
T275		V19D	V21D - - MV01 F2(Y)
T276		V19E	V21E - - MV01 F2(Z) F2(Z.W) -F2(X.Y.Z)
T277		V19F	V21F - - MV01 F2(W)

6.0 ECSSL LISTING

INPUT LISTING

```
      *CONSOLE=681,1
C     EAI 681 ANALOG PROCESSOR COMPONENT DECK                    (MICOM 1)
C
C     COMBINATION INTEGRATOR/SUMMERS, EACH WITH 2 COEFF UNITS IF AVAILABLE
      A000,C000,C002
      A005,C005,C007
      A010,C010,C012
      A015,C015,C017
      A020,C020,C022
      A025,C025,C027
      A030,C030,C032
      A035,C035,C037
      A040,C040,C042
      A045,C045,C047
      A050,C050,C052
      A055,C055,C057
      A060,C060,C062
      A065,C065,C067
      A070,C070,C072
      A075,C075,C077
      A080,C080,C082
      A085,C085,C087
      A090,C090,C092
      A095,C095,C097
      A100,C100,C102
      A105,C105,C107
      A110,C110,C112
      A115,C115,C117
      A002,C004
      A007,C009
      A012,C014
      A017,C019
      A022,C024
      A027,C029
C
C     SUMMERS, EACH WITH 2 COEFF UNITS IF AVAILABLE
      /
      *A001,C001,C003
      A006,C006,C008
      A011,C011,C013
      A016,C016,C018
      A021,C021,C023
      A026,C026,C028
      A031,C031,C033
      A036,C036,C038
      A041,C041,C043
      A046,C046,C048
      A051,C051,C053
      A056,C056,C058
      A061,C061,C063
      A066,C066,C068
      A071,C071,C073
      A076,C076,C078
      A081,C081,C083
      A086,C086,C088
      A091,C091,C093
      A096,C096,C098
      A101,C101,C103
      A106,C106,C108
      A111,C111,C113
```

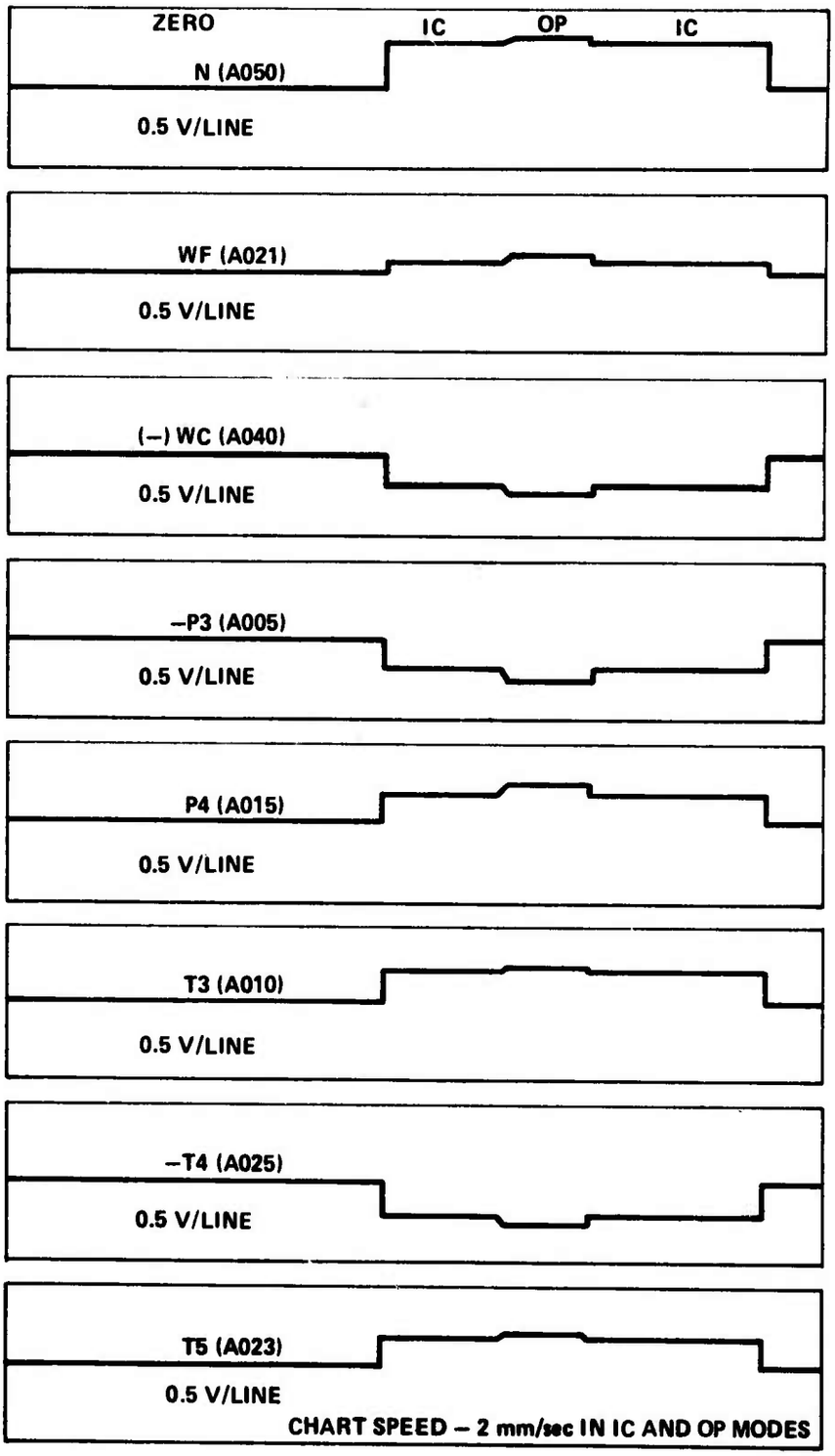
```
      A116,C116,C118
C
C   MULTIPLIERS
      /
     *A003,A008,A033,A038,A013,A018,A043,A048,A023,A028,A053,A058,
     1A088,A083,A118,A113,A078,A073,A108,A103,A068,A063,A098,A093
C
C   COMPARATOR AMPLIFIERS/INVERTERS
      A004,A009,A034,A039,A014,A019,A044,A049,A024,A029,A054,A059,
     1A089,A084,A119,A114,A079,A074,A109,A104,A069,A064,A099,A094
C
C   LOG GENERATORS
      A032,A037,A042,A047,A052,A057
C
C   SINE/COSINE GENERATORS
      A087,A082,A117,A112,A077,A072,A107,A102,A067,A062,A097,A092
C
C   DIGITALLY-CONTROLLED FUNCTION GENERATORS
      F000,F001,F002,F003,F004,F005,F006,F007
      F010,F011,F012,F013,F014,F015,F016,F017
C
C   FREE COEFFICIENTS
      C034,C039,C044,C049,C054,C059,
     1C084,C089,C114,C119,C074,C079,C104,C109,C064,C069,C094,C099
C
C   TRUNKS
      T020,T021,T022,T023,T024,T025,T026,T027,T028,T029,
     1T030,T031,T032,T033,T034,T035,T036,T037,T038,T039,
     2T040,T041,T042,T043,T044,T045,T046,T047,T048,T049,
     3T050,T051,T052,T053,T054,T055,T056,T057,T058,T059,
     4T080,T081,T082,T083,T084,T085,T086,T087,T088,T089,
     5T090,T091,T092,T093,T094,T095,T096,T097,T098,T099
     6T100,T101,T102,T103,T104,T105,T106,T107,T108,T109
     7T110,T111,T112,T113,T114,T115,T116,T117,T118,T119
C
C   LIMITERS \ USE L FOR DIGITALLY-SET , H FOR HAND-SET '
      H0C1,H011,H021,H031,H041,H051,H061,H071,H081,H091,H101,H111
C
C
C   END OF 681 ANALOG PROCESSOR ASSIGNMENT
      *TITLE=
COMPONENTS ACCEPTED -
 30 INTEGRATORS
 24 SUMMERS
 24 Q.S.MULTIPLIERS
 24 COMPARATORS
  6 LOG/EXP DFGS
 12 SIN/COS DFGS
 16 VARIABLE DFGS
 18 INDEPENDENT POTS
 78 TRUNKS
 12 LIMITERS
  0 MVFG MODULES
```

INPUT LISTING (AFTER COMPONENT DECK)

```
                *CONSOLE=681,1
                *TITLE=
CASE NO. 1.0    J65JET
                *INPUT
C
                EQUATIONS
C               ***************************************************
C
                DER(TIME)=1
                P3=P2*(F1(N)+F2(WCF3))
                WCF3=WC-F3(N)
                T3=T2*(F4(N)+F5(WCF6))+T2
                WCF6=WC-F6(N)
                DH3=2.541E-5*(T3**2-T2**2)/2+.2265*(T3-T2)
                P4=.95*P3
                T4=71764.7*F7(P4)*WF/WC+T3
                WCLP=F8(P4N)*P4/SQRT(T4)
                DER(WC)=-1000*WC+1000*WCLP
                DENT=1+(1.2368-.1184E-3*T4)*F9(P4N)
                T5=T4/DENT
                DH4=2.541E-5*(T4**2-T5**2)/2+.2265*(T4-T5)
                NC=NCOM+NCO
                DH43=DH4-DH3
                DER(N)=4721*DH43*WC/N
                DWF=(K1*NE+K2*NEI)/10000
                WF=WFO+DWF
                NECS=LIMIT(-500,500,NC-N)
                DER(NEI)=NE
                DER(NE)=-10*NE+10*NECS
                P4N=P4/P2
C
                CONSTANTS
C               ***************************************************
C
                P2=29.92
                T2=519
                H2=124.03
C
C               CONSTANTS PARAMETRIC
C***************************************************************
C
                NCO=8000,10000,3500
                WFO=.9144,6
                NCOM=0,10000
                K1=1.2,10
                K2=2,10
C
                VARIABLES DIFFERENTIAL
C***************************************************************
                TIME=10,30,^
                N=7000,10000,3500,^
                NE=1000,10000,^
                NEI=1000,10000,^
                WC=93.66,200,20,^
C
C               VARIABLES ALGEBRAIC
C               ***************************************************
                WCF3,140=,^
                WCLP,200=,^
                T3,1500=,500,^
                WCF6,140=,^
                DH3,300=,0,^
```

```
      P3,300=,30,^
      P4,300=,^
      T4,3000=,500,^
      WF,6=,^
      T5,3000=,500,^
      DH4,300=,0,^
      NC,10000,3500,^
      NECS,10000,^
      DWF,6,^
      P4N,10,1,^
      DENT,3=,^
      DH43,300=,^
C
      FUNCTIONS
C     ***************************************************************
C
      DUMN=0,1500,3000,3500,4000,4500,5000,5500,6000,6500,7000,
     1    7500,8000,8300,8600,9500
      DMWCF3=0,5,10,15,20,21,22,23,24,25,26,27,28,29,29.5,30
      DMWCF6=0,3,5,10,15,20,22,24,25,26,27,28,29,29.5,30,30.2
      DMP4=2,3,4,5,7,8,10,12,14,15,20,25,30,100,200,300
      DMP4P2=1.0613,1.1174,1.2363,1.3629,1.4883,1.6172,1.7536,1.9029,
     1      2.0625,2.4159,2.5988,2.8428,3.3288,3.9946,5.1459,9.7810
      F1(DUMN)=1.00,0.97,1.20,1.34,1.54,1.74,2.03,2.39,2.93,3.76,5.28,
     1      6.46,7.33,8.23,9.16,10.00
      F3(DUMN)=0.0,-4.5,-1.4,2.4,6.7,12.7,20.4,29.0,40.0,52.5,66.2,
     1      80.1,91.0,97.5,102.5,108.5
      F4(DUMN)=0.0,.204,.363,.405,.469,.510,.533,.623,.699,.784,.927,
     1      1.03,1.103,1.218,1.339,1.487
      F6(DUMN)=0.0,-7.15,-5.0,-1.0,3.5,10.5,18.5,27.5,38.25,51.2,65.3,
     1    79.2,90.5,96.3,101.2,107.3
      F2(DMWCF3)=0.0,0.07,0.115,0.115,0.09,0.08,0.05,0.0,-0.08,-0.19,
     1    -0.34,-0.58,-1.0,-2.5,-3.25,-4.0
      F5(DMWCF6)=0.0,-0.34,-0.55,-0.1,-0.142,-0.182,-0.197,-0.218,-0.23,
     1    -0.243,-0.261,-0.283,-0.312,-0.34,-0.53,-0.7
      F7(DMP4)=.46,.57,.65,.707,.79,.82,.867,.9,.924,.935,.9666,.984,
     1      .99,.99,.99,.99
      F8(DMP4P2)=11.3,15.4,19.6,22.2,24.0,25.4,26.3,27.0,27.5,28.18,
     1      28.39,28.52,28.71,28.82,28.88,28.90
      F9(DMP4P2)=.0091,.0183,.0357,.0522,.0681,.0831,.0976,.1116,.1258,
     1      .1499,.1617,.1731,.1946,.2146,.2328,.25
      END
      *REDUCE
      *ORDER
      *SCALE
      *STATIC
VARIABLE DENT   USED AS A DIVISOR AND ASSUMED POSITIVE IN VALUE
      *ASSIGN
      *OUTPUT=REDUCT
SCALING OF AAV007 HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF ###### HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF AAV012 HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF ###### HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF AAV017 HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF ###### HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF AAV018 HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
SCALING OF ###### HAS A FORCED RELATIONSHIP TO OTHER SCALINGS
      *END
      *STOP*
```

7.0 DYNAMIC VERIFICATION RESULTS(PLOTS)


ZERO
IC
OP
IC
N (A050)
0.5 V/LINE
WF (A021)
0.5 V/LINE
(–) WC (A040)
0.5 V/LINE
–P3 (A005)
0.5 V/LINE
P4 (A015)
0.5 V/LINE
T3 (A010)
0.5 V/LINE
–T4 (A025)
0.5 V/LINE
T5 (A023)
0.5 V/LINE
CHART SPEED – 2 mm/sec IN IC AND OP MODES

Appendix F. MICOM/AFATL HYBRID SIMULATION FACILITY COMPATIBILITY STUDY

Air Force Armament Laboratory
Armament Development and Test Center
Systems Analysis and Simulation Branch
Eglin AFB FL 32542

R&D Statement of Work
DLMA 76-103
23 Feb 76

MICOM/AFATL HYBRID SIMULATION FACILITY COMPATABILITY STUDY

1.0 Introduction. AFATL and the Army Missile Command (MICOM) have independently developed facilities used for real-time, hybrid computer, hardware-in-the-loop simulations of guided weapon flight performance. The equipment and capabilities of these two facilities differ significantly in many respects. MICOM's facility represents the state-of-the art in advanced simulation facilities. The Army is presently evaluating the feasibility of their approach since many of their techniques and facility subsystems are state-of-the-art and not completely proved in this application.

1.1 The much less sophisticated AFATL facility has proved extremely effective in providing a tool for evaluation of weapon system performance and components. Better understanding of AF development programs has been achieved through this facility. Early identification of potential problem areas and increased success probability of test flights have effected substantial cost savings. While the present AFATL facility has been adequate in the past, upgrades and expansion will be necessary in the future. In fact, due to the increasing scope and volume of Air Force guided weapons development programs, ADTC may some day require a dedicated facility comparable to that at MICOM. It is noted, however, that due to present technological uncertainties, it would not be to the Air Force's advantage to build another MICOM-like facility at this time.

1.2 It is very clear that the Air Force should be working with MICOM for several reasons. First, the Air Force could use the facility to satisfy some near-term simulation requirements. Second, the Air Force could gain the considerable experience and knowledge required to develop an extensive dedicated facility if and when that investment becomes necessary. This study effort will determine the feasibility and associated cost of using the MICOM facility to supplement the simulation capabilities of the AFATL facility.

2.0 Scope. The effort under this statement of Work consists in part of a comparison of the hardware and software used for guided weapon simulations at MICOM and ADTC (including AFATL and TSX). The feasibility, cost, and advantages of hardware and software changes that will increase the commonality between MICOM and ADTC simulation efforts will be determined. Particular emphasis will be placed on the feasibility of using MICOM's Radio Frequency Simulation System (RFSS) in support of AFATL air-to-air guided weapon development programs.

3.0 Background. The MICOM facility has target simulation capabilities in the infrared, optical, and radio frequency bands that significantly surpass the capabilities of AFATL's simulation facility. It would cost approximately $20 million to construct an equivalent radio frequency target simulation facility at AFATL. The extensive MICOM facility was justified on projected tri-service use and the Army is very interested in making its facility available to other users.

3.1 Weapon system analyses can be most effectively performed when supported by a simulation facility that is located near the analysts' place of work. Furthermore, much of the simulation needed in such analyses does not require the use of an extensive MICOM-like facility. Therefore, it would be neither practical nor cost-effective for AFATL to rely solely on MICOM simulation support on any given project. Thus, normally AFATL should use the MICOM facility only if it is practical to conduct some simulations at ADTC and some at MICOM. Under that mode of operation, a critical concern would be the mutual equivalence and ease of implementation of the same simulation using two different sets of hardware and software. This effort will determine what hardware and software changes could be made at AFATL and MICOM to alleviate this concern and make that mode of operation practical. Based on a preliminary exchange of information between these two facilities, it appears that software changes at AFATL offer the most potential for achieving the required degree of simulation compatibility at an acceptable cost.

4.0 Task.

4.1 Capabilities of MICOM Target Simulation Cells. MICOM shall summarize the current and projected capabilities of the MICOM target simulation cells. This capability description shall address both open-loop, cell-only simulation and closed-loop simulation using a target cell in conjunction with the MICOM hybrid computer center. It shall include the cells' capability to simulate the real-time weapon flight dynamics and the relative motion of targets with respect to the weapon. It shall include a functional description of what weapon subsystem data could be acquired during open-loop and closed-loop simulations and shall also describe the functional interface required between the weapon subsystem and the simulation cell. The major emphasis in this task shall be placed on the capabilities of the RFSS cell.

4.2 AFATL Use of MICOM Target Simulation Cells. MICOM shall estimate the cost per day of AFATL use of each of the MICOM target simulation cells. The cost of both open-loop cell-only simulation and closed-loop simulation shall be estimated. MICOM shall also document how scheduling and priority of AFATL use of the cells will be determined.

4.3 Hybrid Computer Compatibility. MICOM shall assist ADTC in determining the feasibility and associated cost of establishing a direct link between the ADTC CDC 6600 and the AFATL hybrid computers in order to achieve a hybrid computer complex at AFATL having a digital processing capability equivalent to the one at MICOM. MICOM shall also determine the practicality and cost of dedicating one of MICOM's Pacer 100 CPU's to hybrid simulation when AFATL is using the MICOM hybrid computer facility. The purchase and installation at MICOM of an additional Pacer 100 CPU and appropriate peripherals and interfacing shall also be considered. The goal of this task shall be the determination of the most cost effective way to achieve easy transfer of the digital portion of a hybrid simulation from AFATL to MICOM.

4.4 AFATL Use of the Automatic Programming and Scaling of Equations (APSE) Compiler. MICOM shall assist ADTC in determining the feasibility and associated cost of obtaining and implementing on the ADTC CDC 6600 the version of the APSE Compiler used by MICOM. If MICOM intends to have future revisions made in this compiler, the cost and possible problems involved in maintaining equivalence between the MICOM and AFATL compilers shall also be addressed. MICOM shall estimate the cost of implementing and checkout on one of MICOM's AD4 analog computers an analog program that had been implemented using the APSE Compiler on one of AFATL's EAI 680 or 681 analog computers. MICOM shall also estimate the cost of this transfer of analog computer programs if AFATL did not adopt the use of the APSE Compiler.

4.5 AFATL Use of the Advanced Continuous Simulation Language (ACSL). MICOM shall assist ADTC in determining the cost of obtaining and implementing on the ADTC CDC 6600 the version of ACSL used by MICOM. If MICOM intends to have revisions made in this language, the cost and possible problems involved in maintaining the necessary equivalence between the MICOM and AFATL versions of this language shall also be addressed.

4.6 Documentation. MICOM shall document the results obtained in this effort. The MICOM simulation facility capabilities and usage costs obtained in Tasks 4.1 and 4.2 will be contained in one report. A second report will contain all of the information obtained in Tasks 4.3, 4.4, and 4.5. This second report will include a recommended set of procedures and hardware and software changes to be made at ADTC and MICOM to achieve the required degree of MICOM/AFATL simulation compatibility.

6.0 Reports, Data, and Other Deliverables. Reports and data shall be prepared and delivered in accordance with attached DD Form 1423.

Appendix G. CALCULATION OF DIGITAL INTERFACE TRANSMISSION TIMES (11 JUNE 1976)

Definitions

CABL	Transmission cable length (ft)
DLWR	Total delay due to wire in each block of data (μsec)
FMTM	Simulation frame time (msec)
GTDL	Gate propagation delay (nsec)
LNST	Line settling time for each transmission (nsec)
NDWB	Number of data words in a data block in one frame
NGAT	Number of gates in transmission path excluding response circuit
NRES	Number of responses required per word transmitted (not necessarily data)
NTRW	Number of transmissions required per word transmitted
NTTX	Number of transmitters plus receivers in path from source to destination
NWTD	Number of words which must be transmitted per data word
PRBK	Percent of time required to transmit one block of data (TTXB) consumed by wire delay
PRFM	Percent of simulation frame time (FMTM) consumed to transmit a block of data with length NDWB
PRTM	Propagation time per foot of cable (nsec)
SUTM	Time required to set up for each frame (nsec)
TRRS	Time for the receiving device to respond to a transmission sent to it by a device requiring a response (nsec)
TTOW	Time required to transmit one word (not necessarily data) (μsec)
TTXB	Total time required to transmit a block of data words (μsec)
TTXD	Total time required to transmit one data word (μsec)
XRDL	Transmitter/receiver propagation delay (nsec)

```
1.005     @T, 0;
1.010     : "CALCULATION OF DIGITAL INTERFACE TRANSMISSION TIMES": :
1.020     "1048 11 JUNE 1976 VERSION": :
1.030
1.040     11;
1.042     "SCAN OR SINGLE CASE(Y OR N)(DEFAULT IS SINGLE CASE)": : SCAN←
1.044     SCAN==1? 2.
1.050     : "ENTER TEST VALUES": :
1.060     CABL← NDWB← FMTM←
1.150     12; : : 13; 14;
1.999     @T, 0; 1.

2.010     '"SCAN PARAMETERS"
2.020     @L, 0; 5; 1.01; 1.02; 5; @T, 0; 11;
2.040     "SCAN DATA BLOCK SIZE", NDWB=DBS[1]
2.042     @L, 0; 13.022; 13.03; 13.04; 13.05; 13.06;
2.050     2.1.
2.060     5; "DATA BLOCK SIZE = ", $NI, $FI, NDWB, $WI, "WORDS(NDWB)": :
2.100     I=1, 1, 3! 2.06; 3; NDWB=DBS[I+1]
2.999     @T, 0; 1.

3.010     "SCAN FRAME TIME", FMTM=FT[1], 3.1.
3.020     "FRAME TIME = ", $NI, $FI, FMTM, $WI, "MILLISECONDS(FMTM)":
3.100     J=1, 1, 6! 3.02; 4; FMTM=FT[J+1]

4.010     "SCAN CABLE LENGTH"
4.020     13.13; "PUT OUT HEADER"
4.030     CABL=CAB[1]
4.100     K=1, 1, 4! 12; 14; CABL=CAB[K+1]
4.999     : : 11.07; "RESET CABL"

5.010     "------------------------------------------------------------------------

11.010    "DATA TABLE"
11.020    GTDL=15, LNST=20
11.030    NGAT=2, NPSS=2
11.040    NTRN=4, NTTX=2
11.050    NWTD=2, PRTM=2
11.060    TPRS=15, MRDL=50
11.070    CABL=0
11.072    NDWB=30, SUTM=0
11.080    FMTM=1
11.090    SCAN=0
11.100    "DATA BLOCK SIZE (WORDS) ARRAY"
11.110    DBS[1]=1, DBS[2]=10, DBS[3]=100, DBS[4]=0
11.120    "FRAME TIME (MS) ARRAY"
11.130    FT[1]=1, FT[2]=5, FT[3]=10
11.140    FT[4]=20, FT[5]=50, FT[6]=100, FT[7]=0
11.150    Y=1, N=0
11.160    "CABLE LENGTH ARRAY"
11.170    CAB[1]=0, CAB[2]=100, CAB[3]=500
11.180    CAB[4]=1000, CAB[5]=0

12.010    "CALCULATIONS"
12.020    DLWR=NDWB*NWTD*NTRN*PRTM*CABL/1000
12.030    TTDN=(NTRN*(NTTX*(MRDL+NGAT*GTDL+LNST+PRTM*CABL)+NRES*TPRS)/1000
12.040    TTXD=TTDN*NWTD
12.050    TTXB=TTXD*NDWB+SUTM
12.060    PRFM=(TTXB)/(FMTM*10)
12.070    PPBK=(DLWR*100)/TTXB
```

```
13.010      "OUTPUT HEADER INFO"
13.020      @L, 0; :
13.022      "CONSTANTS AND INITIAL CONDITIONS": :
13.030      GTDL, LNST, NGAT, NPES:
13.040      NTRW, NTTX, NWTD, PRTM:
13.050      TRRS, XRDL, CABL, NDWB:
13.060      FNTM, SUTM: :
13.130      : "   DLWR      TTDW      TTXD      TTXB      PRFM      PRBK      CABL":
13.999      @T, 0;

14.010      "OUTPUT DATA"
14.012      @L, 0;
14.015
14.020      $NI, DLWR, TTDW, TTXD, TTXB, PRFM, PRBK, CABL:
14.999

15.010      "LIST ON GOULD"
15.020      @L, 0; $CLOSE: $PAR: @T, 0;

16.010      "PUNCH PAPER TAPE"
16.020      @P, 0; $PAR: $END: $CLOSE:

17.010      "DUMP TO #2 MAG TAPE"
17.020      @DEL, , @M1, 0; $PAR: $END: $CLOSE:
```

CALCULATION OF DIGITAL INTERFACE TRANSMISSION TIMES

1019 11 JUNE 1976 VERSION

CONSTANTS AND INITIAL CONDITIONS

GTDL = 15.0000, LNST = 20.0000, NGAT = 2.00000, NRES = 2.00000
NTRN = 4.00000, NTTX = 2.00000, NWTD = 2.00000, PRTM = 2.00000
TRRS = 15.0000, XRDL = 50.0000, CABL = .000000, NDWB = 1.00000
FMTM = 1.00000, SUTM = .000000

DATA BLOCK SIZE = , 1 , WORDS(NDWB)

FRAME TIME = , 1 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.126000,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.286000,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.926000,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	1.72600,	92.6999,	1000.00

FRAME TIME = , 5 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.025200,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.057200,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.185200,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	.345200,	92.6999,	1000.00

FRAME TIME = , 10 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.012600,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.028600,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.092600,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	.172600,	92.6999,	1000.00

FRAME TIME = , 20 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.006300,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.014300,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.046300,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	.086300,	92.6999,	1000.00

FRAME TIME = , 50 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.002520,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.005720,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.018520,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	.034520,	92.6999,	1000.00

FRAME TIME = , 100 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	1.26000,	.001260,	.000000,	.000000
1.60000,	1.43000,	2.86000,	2.86000,	.002860,	55.9440,	100.000
8.00000,	4.63000,	9.26000,	9.26000,	.009260,	86.3931,	500.000
16.0000,	8.63000,	17.2600,	17.2600,	.017260,	92.6999,	1000.00

DATA BLOCK SIZE = , 10 , WORDS(NDWB)

FRAME TIME = , 1 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	12.6000,	1.26000,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	2.86000,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	9.26000,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	17.2600,	92.6999,	1000.00

FRAME TIME = , 5 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	12.6000,	.252000,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	.572000,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	1.85200,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	3.45200,	92.6999,	1000.00

FRAME TIME = , 10 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	12.6000,	.126000,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	.286000,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	.926000,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	1.72600,	92.6999,	1000.00

FRAME TIME = , 20 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	12.6000,	.063000,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	.143000,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	.463000,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	.863000,	92.6999,	1000.00

FRAME TIME = , 50 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	12.6000,	.025200,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	.057200,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	.185200,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	.345200,	92.6999,	1000.00

FRAME TIME = , 100 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL

.000000,	.630000,	1.26000,	12.6000,	.012600,	.000000,	.000000
16.0000,	1.43000,	2.86000,	28.6000,	.028600,	55.9441,	100.000
80.0000,	4.63000,	9.26000,	92.6000,	.092600,	86.3931,	500.000
160.000,	8.63000,	17.2600,	172.600,	.172600,	92.6999,	1000.00

DATA BLOCK SIZE = , 100 , WORDS(NDWB)

FRAME TIME = , 1 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	12.6000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	28.6000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	92.6000,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	172.600,	92.6999,	1000.00

FRAME TIME = , 5 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	2.52000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	5.72000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	18.5200,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	34.5200,	92.6999,	1000.00

FRAME TIME = , 10 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	1.26000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	2.86000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	9.26000,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	17.2600,	92.6999,	1000.00

FRAME TIME = , 20 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	.630000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	1.43000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	4.63000,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	8.63000,	92.6999,	1000.00

FRAME TIME = , 50 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	.252000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	.572000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	1.85200,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	3.45200,	92.6999,	1000.00

FRAME TIME = , 100 , MILLISECONDS(FMTM)

DLWR	TTOW	TTXD	TTXB	PRFM	PRBK	CABL
.000000,	.630000,	1.26000,	126.000,	.126000,	.000000,	.000000
160.000,	1.43000,	2.86000,	286.000,	.286000,	55.9441,	100.000
800.000,	4.63000,	9.26000,	926.000,	.926000,	86.3931,	500.000
1600.00,	8.63000,	17.2600,	1726.00,	1.72600,	92.6999,	1000.00

DISTRIBUTION

	No. of Copies
Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
Commander US Army Materiel Command	
Attn: DRCRD	1
DRCDL	1
5001 Eisenhower Avenue Alexandria, Virginia 22333	
Air Force Armament Laboratory Attn: Major Radoy Eglin Air Force Base, Florida 32544	10
DRSMI-FR, Mr. Strickland	1
-LP, Mr. Voigt	1
DRDMI-X, Dr. McDaniel	1
-T, Dr. Kobler	1
-TD	10
-TB	3
-TI (Record Set)	1
(Reference Copy)	1